# PP-Module for

Version: 1.2
2023-06-30
**National Information Assurance Partnership**

## Revision History

| Version | Date | Comment |
|---|---|---|
| 1.3 | 2023-06-30 | Incorporation of NIAP Technical Decisions, TC feedback |
| 1.2 | 2022-03-31 | Format conversion, incorporation of NIAP Technical Decisions, TC feedback |
| 1.1 | 2020-06-18 | Compatibility with CPP_ND_V2.2E, incorporation of NIAP Technical Decisions |
| 1.0 | 2019-09-17 | Initial publication |

## Contents


1 Introduction
1.1 Overview
1.2 Terms
1.2.1 Common Criteria Terms
1.2.2 Technical Terms
1.3 Compliant Targets of Evaluation
1.3.1 TOE Boundary
1.4 Use Cases
2 Conformance Claims
3 Security Problem Description
3.1 Threats
3.2 Assumptions
3.3 Organizational Security Policies
4 Security Objectives
4.1 Security Objectives for the TOE
4.2 Security Objectives for the Operational Environment
4.3 Security Objectives Rationale
5 Security Requirements
5.1 NDcPP Security Functional Requirements Direction
5.1.1 Modified SFRs
5.1.1.1 Cryptographic Support (FCS)
5.1.1.2 Identification and Authentication (FIA)
5.1.1.3 Security Management (FMT)
5.1.1.4 Protection of the TSF (FPT)
5.2 TOE Security Functional Requirements
5.2.1 Auditable Events for Mandatory SFRs
5.2.2 Security Audit (FAU)
5.2.3 Cryptographic Support (FCS)
5.2.4 Security Management (FMT)
5.2.5 Packet Filtering (FPF)
5.2.6 Protection of the TSF (FPT)
5.2.7 Trusted Path/Channels (FTP)
5.3 TOE Security Functional Requirements Rationale
5.4 TOE Security Assurance Requirements
6 Consistency Rationale
6.1 collaborative Protection Profile for Network Devices
6.1.1 Consistency of TOE Type
6.1.2 Consistency of Security Problem Definition
6.1.3 Consistency of Objectives
6.1.4 Consistency of Requirements
Appendix A - Optional SFRs
A.1 Strictly Optional Requirements
A.1.1 Auditable Events for Strictly Optional SFRs
A.1.2 Packet Filtering (FPF)
A.2 Objective Requirements
A.3 Implementation-based Requirements
A.3.1 Auditable Events for Implementation-Dependent SFRs
A.3.2 TOE Access (FTA)
Appendix B - Selection-based Requirements
B.1 Auditable Events for Selection-based SFRs
B.2 Cryptographic Support (FCS)
B.3 Identification and Authentication (FIA)
Appendix C - Extended Component Definitions
C.1 Extended Components Table
C.2 Extended Component Definitions
C.2.1 Packet Filtering (FPF)
C.2.1.1 FPF_RUL_EXT Packet Filtering Rules
C.2.1.2 FPF_MFA_EXT Multifactor Authentication Filtering

C.2.2 Protection of the TSF (FPT)
C.2.2.1 FPT_TST_EXT TSF Self-Test
C.2.3 Cryptographic Support (FCS)
C.2.3.1 FCS_EAP_EXT EAP-TLS
C.2.4 Identification and Authentication (FIA)
C.2.4.1 FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys
C.2.4.2 FIA_PSK_EXT Pre-Shared Key Composition
C.2.4.3 FIA_TOTP_EXT Time-Based One-Time Password Pre-Shared Keys
C.2.5 TOE Access (FTA)
C.2.5.1 FTA_VCM_EXT VPN Client Management
Appendix D - Implicitly Satisfied Requirements
Appendix E - Entropy Documentation and Assessment
Appendix F - Acronyms
Appendix G - Bibliography

# 1 Introduction

## 1.1 Overview

The scope of this Protection Profile Module (PP-Module) is to describe the security functionality of a virtual private network (VPN) gateway in terms of [CC] and to define functional and assurance requirements for such products. This PP-Module is intended for use with the following Base-PPs:

- collaborative Protection Profile for Network Devices, Version 2.2E (NDcPP or CPP_ND_V2.2E)

This Base-PP is valid because a VPN gateway is a device at the edge of a private network that terminates an IPsec tunnel, which provides device authentication, confidentiality, and integrity of information traversing a public or untrusted network. This is functionality that typically will be implemented by a network device.
A Target of Evaluation (TOE) that conforms to a PP-Configuration containing this PP-Module may be a 'Distributed TOE' as defined in the NDcPP; however, the VPN gateway functionality described in this PP-Module should be in a single TOE component. This PP-Module does not prohibit the TOE from implementing other security functionality in a distributed manner. For example, a TOE may have a centralized device that performs VPN gateway and other security functionality (such as intrusion prevention) with a number of distributed nodes that help in the enforcement of the secondary functionality.

## 1.2 Terms

The following sections list Common Criteria and technology terms used in this document.

### 1.2.1 Common Criteria Terms

| | |
|---|---|
| Assurance | Grounds for confidence that a TOE meets the SFRs [CC]. |
| Base Protection Profile (Base-PP) | Protection Profile used as a basis to build a PP-Configuration. |
| Collaborative Protection Profile (cPP) | A Protection Profile developed by international technical communities and approved by multiple schemes |
| Common Criteria (CC) | Common Criteria for Information Technology Security Evaluation (International Standard ISO/IEC 15408). |
| Common Criteria Testing Laboratory | Within the context of the Common Criteria Evaluation and Validation Scheme (CCEVS), an IT security evaluation facility, accredited by the National Voluntary Laboratory Accreditation Program (NVLAP) and approved by the NIAP Validation Body to conduct Common Criteria-based evaluations. |
| Common Evaluation Methodology (CEM) | Common Evaluation Methodology for Information Technology Security Evaluation. |
| Distributed TOE | A TOE composed of multiple components operating as a logical whole. |
| Extended Package (EP) | A deprecated document form for collecting SFRs that implement a particular protocol, technology, or functionality. See Functional Packages. |
| Functional Package (FP) | A document that collects SFRs for a particular protocol, technology, or functionality. |
| Operational Environment (OE) | Hardware and software that are outside the TOE boundary that support the TOE functionality and security policy. |
| Protection Profile (PP) | An implementation-independent set of security requirements for a category of products. |
| Protection Profile Configuration (PP-Configuration) | A comprehensive set of security requirements for a product type that consists of at least one Base-PP and at least one PP-Module. |
| Protection Profile Module | An implementation-independent statement of security needs for a TOE type complementary to one or more Base Protection Profiles. |

| | |
|---|---|
| (PP-Module) | |
| Security Assurance Requirement (SAR) | A requirement to assure the security of the TOE. |
| Security Functional Requirement (SFR) | A requirement for security enforcement by the TOE. |
| Security Target (ST) | A set of implementation-dependent security requirements for a specific product. |
| Target of Evaluation (TOE) | The product under evaluation. |
| TOE Security Functionality (TSF) | The security functionality of the product under evaluation. |
| TOE Summary Specification (TSS) | A description of how a TOE satisfies the SFRs in an ST. |

### 1.2.2 Technical Terms

| | |
|---|---|
| Headend | A VPN use case where the VPN gateway is establishing VPN connectivity with endpoint VPN clients as opposed to other infrastructure devices (e.g. site-to-site). |
| Packet Filtering | The process by which an edge network device determines if traffic bound to or from its external network is passed to its destination or dropped. |
| VPN Gateway | A type of network device that resides at the edge of a private network and permits the establishment of VPN connectivity from computers residing in an external network. |
| Virtual Private Network (VPN) | A mechanism for overlaying a cryptographically secured network over distributed wide-area networks. |

## 1.3 Compliant Targets of Evaluation

This PP-Module specifically addresses network gateway devices that terminate IPsec VPN tunnels. A compliant VPN gateway is a device composed of hardware and software that is connected to two or more distinct networks and has an infrastructure role in the overall enterprise network. In particular, a VPN gateway establishes a secure tunnel that provides an authenticated and encrypted path to one or more other sites and thereby decreases the risk of exposure of information transiting an untrusted network.
The baseline requirements of this PP-Module are those determined necessary for a multi-site VPN gateway device. A compliant TOE may also contain the ability to act as a headend for remote clients. Because this capability is optional, the remote client-based requirements have been included within Appendix A.

### 1.3.1 TOE Boundary

The physical boundary for a TOE that conforms to this PP-Module is a hardware appliance that also provides generalized network device functionality, such as auditing, identification and authentication, and cryptographic services for network communications. The TOE’s logical boundary includes all functionality required by the claimed Base-PP as well as the VPN functionality and related capabilities that are defined in this PP-Module. Any functionality that is provided by the network device that is not relevant to the security requirements defined by this PP-Module or the Base-PP is considered to be outside the scope of the TOE.

## 1.4 Use Cases

This PP-Module defines two potential use cases for the VPN gateway TOE, defined below. The first use case will always be applicable for a TOE that conforms to this PP-Module. The second use case defines an optional deployment for the TOE that accompanies the first use case.

**[USE CASE 1] Network Device**

The VPN gateway is part of the functionality that is provided by a general network device appliance, such as a router or switch, or a device that is dedicated solely to providing multi-site VPN gateway functionality.

**[USE CASE 2] Remote Client Headend**

The VPN gateway provides the ability to act as a headend for remote clients.

# 2 Conformance Claims

### Conformance Statement

This PP-Module inherits exact conformance as required from the specified Base-PPs and as defined in the CC and CEM addenda for Exact Conformance, Selection-Based SFRs, and Optional SFRs (dated May 2017).

The following PPs and PP-Modules are allowed to be specified in a PP-Configuration with this PP-Module:

- collaborative Protection Profile Module for Stateful Traffic Filter Firewalls v1.4 + Errata 20200625
- PP-Module for Intrusion Protection Systems, v1.0

### CC Conformance Claims

This PP-Module is conformant to Parts 2 (extended) and 3 (conformant) of Common Criteria Version 3.1, Revision 5 [CC].

### PP Claim

This PP-Module does not claim conformance to any PP.

### Package Claim

This PP-Module does not claim conformance to any packages.

# 3 Security Problem Description

The security problem is described in terms of the threats that the TOE is expected to address, assumptions about its operational environment (OE), and any organizational security policies that the TOE is expected to enforce.

## 3.1 Threats

The following threats defined in this PP-Module extend the threats defined by the Base-PP.

**T.DATA_INTEGRITY**

Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to modify the data without authorization. If known malicious external devices are able to communicate with devices on the protected network or if devices on the protected network can communicate with those external devices then the data contained within the communications may be susceptible to a loss of integrity.

**T.NETWORK_ACCESS**

Devices located outside the protected network may seek to exercise services located on the protected network that are intended to only be accessed from inside the protected network or only accessed by entities using an authenticated path into the protected network. Devices located outside the protected network may, likewise, offer services that are inappropriate for access from within the protected network.

From an ingress perspective, VPN gateways can be configured so that only those network servers intended for external consumption by entities operating on a trusted network (e.g., machines operating on a network where the peer VPN gateways are supporting the connection) are accessible and only via the intended ports. This serves to mitigate the potential for network entities outside a protected network to access network servers or services intended only for consumption or access inside a protected network.

From an egress perspective, VPN gateways can be configured so that only specific external services (e.g., based on destination port) can be accessed from within a protected network, or moreover are accessed via an encrypted channel. For example, access to external mail services can be blocked to enforce corporate policies against accessing uncontrolled email servers, or, that access to the mail server must be done over an encrypted link.

**T.NETWORK_DISCLOSURE**

Devices on a protected network may be exposed to threats presented by devices located outside the protected network, which may attempt to conduct unauthorized activities. If known malicious external devices are able to communicate with devices on the protected network, or if devices on the protected network can establish communications with those external devices (e.g., as a result of a phishing episode or by inadvertent responses to email messages), then those internal devices may be susceptible to the unauthorized disclosure of information.

From an infiltration perspective, VPN gateways serve not only to limit access to only specific destination network addresses and ports within a protected network, but whether network traffic will be encrypted or transmitted in plaintext. With these limits, general network port scanning can be prevented from reaching protected networks or machines, and access to information on a protected network can be limited to that obtainable from specifically configured ports on identified network nodes (e.g., web pages from a designated corporate web server). Additionally, access can be limited to only specific source addresses and ports so that specific networks or network nodes can be blocked from accessing a protected network thereby further limiting the potential disclosure of information.

From an exfiltration perspective, VPN gateways serve to limit how network nodes operating on a protected network can connect to and communicate with other networks limiting how and where they can disseminate information. Specific external networks can be blocked altogether or egress could be limited to specific addresses or ports. Alternately, egress options available to network nodes on a protected network can be carefully managed in order to, for example, ensure that outgoing connections are encrypted to further mitigate inappropriate disclosure of data through packet sniffing.

**T.NETWORK_MISUSE**

Devices located outside the protected network, while permitted to access particular public services offered inside the protected network, may attempt to conduct inappropriate activities while communicating with those allowed public services. Certain services offered from within a protected network may also represent a risk when accessed from outside the protected network.

From an ingress perspective, it is generally assumed that entities operating on external networks are not bound by the use policies for a given protected network. Nonetheless, VPN gateways can log policy violations that might indicate violation of publicized usage statements for publicly available services.

From an egress perspective, VPN gateways can be configured to help enforce and monitor protected network use policies. As explained in the other threats, a VPN gateway can serve to limit dissemination of data, access to external servers, and even disruption of services – all of these could be related to the use policies of a protected network and as such are subject in some regards to enforcement. Additionally, VPN gateways can be configured to log network usages that cross between protected and external networks and as a result can serve to identify potential usage policy violations.

**T.REPLAY_ATTACK**

If an unauthorized individual successfully gains access to the system, the adversary may have the

opportunity to conduct a “replay” attack. This method of attack allows the individual to capture packets traversing throughout the network and send the packets at a later time, possibly unknown by the intended receiver. Traffic is subject to replay if it meets the following conditions:

- Cleartext: an attacker with the ability to view unencrypted traffic can identify an appropriate segment of the communications to replay as well in order to cause the desired outcome
- No integrity: alongside cleartext traffic, an attacker can make arbitrary modifications to captured traffic and replay it to cause the desired outcome if the recipient has no means to detect these

## 3.2 Assumptions

These assumptions are made on the Operational Environment (OE) in order to be able to ensure that the security functionality specified in the PP-Module can be provided by the TOE. If the TOE is placed in an OE that does not meet these assumptions, the TOE may no longer be able to provide all of its security functionality. This PP-Module defines assumptions that extend those defined in the supported Base-PP.
All assumptions for the OE of the Base-PP also apply to this PP-Module. A.NO_THRU_TRAFFIC_PROTECTION is still operative, but only for the interfaces in the TOE that are defined by the Base-PP and not the PP-Module.

**A.CONNECTIONS**

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

## 3.3 Organizational Security Policies

An organization deploying the TOE is expected to satisfy the organizational security policy listed below in addition to all organizational security policies defined by the claimed Base-PP.

This document does not define any additional OSPs.

# 4 Security Objectives

## 4.1 Security Objectives for the TOE

**O.ADDRESS_FILTERING**

To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption or denial of services, and network-based reconnaissance, compliant TOE's will implement packet filtering capability. That capability will restrict the flow of network traffic between protected networks and other attached networks based on network addresses of the network nodes originating (source) or receiving (destination) applicable network traffic as well as on established connection information.
Addressed by: FPF_RUL_EXT.1, FTA_VCM_EXT.1 (optional)

**O.AUTHENTICATION**

To further address the issues associated with unauthorized disclosure of information, a compliant TOE's authentication ability (IPSec) will allow a VPN peer to establish VPN connectivity with another VPN peer and ensure that any such connection attempt is both authenticated and authorized. VPN endpoints authenticate each other to ensure they are communicating with an authorized external IT entity.
Addressed by: FCS_IPSEC_EXT.1 (refined from Base-PP), FIA_X509_EXT.1/Rev (from Base-PP), FIA_X509_EXT.2 (refined from Base-PP), FIA_X509_EXT.3 (from Base-PP), FTP_ITC.1/VPN, FPF_MFA_EXT.1 (optional), FTA_SSL.3/VPN (optional), FTA_TSE.1 (optional), FCS_EAP_EXT.1 (selection-based), FIA_HOTP_EXT.1 (selection-based), FIA_PSK_EXT.1 (selection-based), FIA_PSK_EXT.2 (selection-based), FIA_PSK_EXT.3 (selection-based), FIA_TOTP_EXT.1 (selection-based)

**O.CRYPTOGRAPHIC_FUNCTIONS**

To address the issues associated with unauthorized disclosure of information, inappropriate access to services, misuse of services, disruption of services, and network-based reconnaissance, compliant TOE's will implement cryptographic capabilities. These capabilities are intended to maintain confidentiality and allow for detection and modification of data that is transmitted outside of the TOE.
Addressed by: FCS_COP.1/DataEncryption (refined from Base-PP), FCS_IPSEC_EXT.1 (refined from Base-PP), FCS_CKM.1/IKE, FCS_EAP_EXT.1 (selection-based), FIA_PSK_EXT.1 (selection-based), FIA_PSK_EXT.2 (selection-based)

**O.FAIL_SECURE**

There may be instances where the TOE's hardware malfunctions or the integrity of the TOE's software is compromised, the latter being due to malicious or non-malicious intent. To address the concern of the TOE operating outside of its hardware or software specification, the TOE will shut down upon discovery of a problem reported via the self-test mechanism and provide signature-based validation of updates to the TSF.
Addressed by: FPT_TST_EXT.1 (refined from Base-PP), FPT_TUD_EXT.1 (refined from Base-PP), FPT_FLS.1/SelfTest, FPT_TST_EXT.3

**O.PORT_FILTERING**

To further address the issues associated with unauthorized disclosure of information, etc., a compliant TOE's port filtering capability will restrict the flow of network traffic between protected networks and other attached networks based on the originating (source) or receiving (destination) port (or service) identified in the network traffic as well as on established connection information.
Addressed by: FPF_RUL_EXT.1

**O.SYSTEM_MONITORING**

To address the issues of administrators being able to monitor the operations of the VPN gateway, it is necessary to provide a capability to monitor system activity. Compliant TOEs will implement the ability to log the flow of network traffic. Specifically, the TOE will provide the means for administrators to configure packet filtering rules to 'log' when network traffic is found to match the configured rule. As a result, matching a rule configured to 'log' will result in informative event logs whenever a match occurs. In addition, the establishment of security associations (SAs) is auditable, not only between peer VPN gateways, but also with certification authorities (CAs).
Addressed by: FAU_GEN.1/VPN, FPF_RUL_EXT.1

**O.TOE_ADMINISTRATION**

TOEs will provide the functions necessary for an administrator to configure the packet filtering rules, as well as the cryptographic aspects of the IPsec protocol that are enforced by the TOE.
Addressed by: FMT_MTD.1/CryptoKeys (refined from Base-PP), FMT_SMF.1/VPN

## 4.2 Security Objectives for the Operational Environment

The OE of the TOE implements technical and procedural measures to assist the TOE in correctly providing its security functionality (which is defined by the security objectives for the TOE). The security objectives for the OE consist of a set of statements describing the goals that the OE should achieve. This section defines the security objectives that are to be addressed by the IT domain or by non-technical or procedural means. The assumptions identified in Section 3 are incorporated as security objectives for the environment. This PP-Module defines environmental security objectives that extend those defined in the supported Base-PP.
All objectives for the OE of the Base-PP also apply to this PP-Module. OE.NO_THRU_TRAFFIC_PROTECTION is still operative, but only for the interfaces in the TOE that are defined by the Base-PP and not the PP-

Module.

**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.

## 4.3 Security Objectives Rationale

This section describes how the assumptions, threats, and organizational security policies map to the security objectives.

**Table 1: Security Objectives Rationale**

| Threat, Assumption, or OSP | Security Objectives | Rationale |
|---|---|---|
| T.DATA_INTEGRITY | O.ADDRESS_FILTERING | The TOE’s ability to provide address filtering helps mitigate the threat of data integrity violations by reducing the amount of potentially malicious network traffic that could potentially exploit the threat. |
| | O.AUTHENTICATION | The TOE’s ability to authenticate entities requesting network access helps mitigate the threat of integrity violations by establishing or exchanging keys that are used to maintain data integrity. |
| | O.CRYPTOGRAPHIC_FUNCTIONS | The modification of data without authorization can be prevented by cryptography that ensures the confidentiality and integrity of the data. |
| | O.FAIL_SECURE | The TOE's ability to protect against unauthorized modifications to itself helps ensure that its functionality cannot be altered in such a way that it fails to maintain integrity of its communications. |
| | O.PORT_FILTERING | The TOE’s ability to provide port filtering helps mitigate the threat of data integrity violations by reducing the amount of potentially malicious network traffic that could potentially exploit the threat. |
| T.NETWORK_ACCESS | O.ADDRESS_FILTERING | The TOE’s address filtering capability helps mitigate the threat of network access by limiting unauthorized reconnaissance activities that can be performed outside the protected network. |
| | O.AUTHENTICATION | The TOE’s ability to authenticate entities requesting network access mitigates unauthorized network access by ensuring that unauthenticated connections cannot access the protected network. |
| | O.CRYPTOGRAPHIC_FUNCTIONS | The TOE’s use of cryptography prevents unauthorized network access by encrypting data transmitted to and from an entity on an untrusted network that is accessing a protected resource. |
| | O.FAIL_SECURE | The TOE's ability to protect against unauthorized modifications to itself helps ensure that its functionality cannot be altered in such a way that it permits network access that it would ordinarily disallow. |
| | O.PORT_FILTERING | The TOE’s port filtering capability helps mitigate the threat of network access by limiting unauthorized reconnaissance activities that can be performed outside the protected network. |
| T.NETWORK_DISCLOSURE | O.ADDRESS_FILTERING | The TOE’s address filtering capability helps mitigate the threat of network disclosure by limiting unauthorized reconnaissance activities that can be performed outside the protected network. |
| | O.FAIL_SECURE | The TOE's ability to protect against unauthorized modifications to itself helps ensure that its functionality cannot be altered to allow unauthorized disclosure of protected network traffic. |
| | O.PORT_FILTERING | The TOE’s port filtering capability helps mitigate the threat of network access by limiting unauthorized reconnaissance activities that can be performed outside the protected network. |
| T.NETWORK_ | O.ADDRESS_ | The TOE’s ability to provide address filtering helps mitigate the |

| | | |
|---|---|---|
| MISUSE | FILTERING | threat of network misuse by reducing the amount of potentially malicious network traffic that could potentially exploit the threat. |
| | O.CRYPTOGRAPHIC_ FUNCTIONS | The TOE’s use of cryptography prevents network misuse by ensuring that an unauthorized attacker cannot inject their own actions into the protected network. |
| | O.FAIL_SECURE | The TOE's ability to protect against unauthorized modifications to itself helps ensure that its functionality to detect potential misuse of network resources is not compromised. |
| | O.PORT_FILTERING | The TOE’s ability to provide port filtering helps mitigate the threat of network misuse by reducing the amount of potentially malicious network traffic that could potentially exploit the threat. |
| | O.SYSTEM_ MONITORING | The TOE’s system monitoring function helps mitigate the threat of network misuse by providing a method to detect when potential misuse is occurring. |
| | O.TOE_ ADMINISTRATION | The TOE implements a management interface that allows for authorized usage of the TOE so that unprivileged users do not have the ability to misuse its functions. |
| T.REPLAY_ ATTACK | O.AUTHENTICATION | The TOE’s ability to enforce authentication helps mitigate replay attacks by making it more difficult for an attacker to impersonate a valid entity. |
| | O.CRYPTOGRAPHIC_ FUNCTIONS | The TOE’s use of cryptography prevents replay attacks by ensuring that network data that is modified and retransmitted will not be parsed as valid traffic. |
| | O.FAIL_SECURE | The TOE's ability to protect against unauthorized modifications to itself helps ensure that it always enforces requirements for confidentiality and integrity of network traffic in the intended manner. |
| A.CONNECTIONS | OE.CONNECTIONS | The OE objective OE.CONNECTIONS is realized through A.CONNECTIONS. |

# 5 Security Requirements

This chapter describes the security requirements which have to be fulfilled by the product under evaluation. Those requirements comprise functional components from Part 2 and assurance components from Part 3 of [CC]. The following conventions are used for the completion of operations:

- **Refinement** operation (denoted by **bold text** or ~~strikethrough text~~): is used to add details to a requirement (including replacing an assignment with a more restrictive selection) or to remove part of the requirement that is made irrelevant through the completion of another operation, and thus further restricts a requirement.
- **Selection** (denoted by *italicized text*): is used to select one or more options provided by the [CC] in stating a requirement.
- **Assignment** operation (denoted by *italicized text*): is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets indicates assignment.
- **Iteration** operation: is indicated by appending the SFR name with a slash and unique identifier suggesting the purpose of the operation, e.g. "/EXAMPLE1."

## 5.1 NDcPP Security Functional Requirements Direction

In a PP-Configuration that includes the NDcPP, the VPN gateway is expected to rely on some of the security functions implemented by the network device as a whole and evaluated against the Base-PP. In this case, the following sections describe any modifications that the ST author must make to the SFRs defined in the Base-PP in addition to what is mandated by section 5.2.

### 5.1.1 Modified SFRs

The SFRs listed in this section are defined in the NDcPP and relevant to the secure operation of the TOE.

#### 5.1.1.1 Cryptographic Support (FCS)

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

FCS_COP.1.1/DataEncryption

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

**Application Note:** This SFR has been modified from its definition in the NDcPP to support this PP-Module's IPsec requirements by mandating support for at least one of CBC or GCM modes and at least one of 128-bit or 256-bit key sizes at minimum. Other selections may be made by the ST author but they are not required for conformance to this PP-Module.

**FCS_IPSEC_EXT.1 IPsec Protocol**

FCS_IPSEC_EXT.1.1

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

**Application Note:** This element is unchanged from its definition in the Base-PP.

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.

**Application Note:** This element is unchanged from its definition in the Base-PP.

FCS_IPSEC_EXT.1.3

The TSF shall implement [**selection**: *transport mode*, *tunnel mode*].

**Application Note:** The selection of supported modes is expected to be performed according to RFC 4301.

FCS_IPSEC_EXT.1.4

The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms **[selection: *AES-CBC-128*, *AES-CBC-256 (specified in RFC 3602)*, *AES-GCM-128*, *AES-GCM-256 (specified in RFC 4106)*] and [selection: *AES-CBC-192 (specified in RFC 3602)*, *AES-GCM-192 (specified in RFC 4106)*, *no other algorithm*]** together with a Secure Hash Algorithm (SHA)-based HMAC [**selection**: *HMAC-SHA-1*, *HMAC-SHA-256*, *HMAC-SHA-384*, *HMAC-SHA-512*, *no HMAC algorithm*].

**Application Note:** This element has been modified from its definition in the NDcPP by mandating either 128 or 256 bit key sizes for AES-CBC or AES-GCM, thereby disallowing for the sole selection of 192 bit key sizes.
When an AES-CBC algorithm is selected, at least one SHA-based HMAC must also be chosen. If only an AES-GCM algorithm is selected, then a SHA-based HMAC is not required since AES-GCM satisfies both confidentiality and integrity functions. IPsec may use a truncated version of the SHA-based HMAC functions contained in the selections. Where a truncated output is used, this is described in the TSS.

FCS_IPSEC_EXT.1.5

The TSF shall implement the protocol: [**selection**:

- *IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [**selection**: no other RFCs for extended sequence numbers, RFC 4304 for extended sequence numbers] and [**selection**: no other RFCs for hash functions, RFC 4868 for hash functions],*
- *IKEv2 as defined in RFC 5996 and [**selection, choose one of**: with no support for NAT traversal, with mandatory support for NAT traversal as specified in RFC 5996, section 2.23] and [**selection**: no other RFCs for hash functions, RFC 4868 for hash functions]*

].

**Application Note:** This element is unchanged from its definition in the Base-PP.

FCS_IPSEC_EXT.1.6

The TSF shall ensure the encrypted payload in the [**selection**: *IKEv1*, *IKEv2*] protocol uses the cryptographic algorithms [**selection**: *AES-CBC-128*, *AES-CBC-256 (specified in RFC 3602)*, *AES-GCM-128*, *AES-CBC-256 (specified in RFC 5282)*].

**Application Note:** This element is changed from its definition in the Base-PP to remove AES-192, which is not recommended. AES-CBC implementation for IPsec is specified in RFC 3602. AES-GCM implementation for IPsec is specified in RFC 5282.

FCS_IPSEC_EXT.1.7

The TSF shall ensure that [**selection**:

- *IKEv1 Phase 1 SA lifetimes can be configured by a Security Administrator based on [**selection**:*
  - *number of bytes,*
  - *length of time, where the time values can be configured within [**assignment**: integer range including 24] hours*

  *],*
- *IKEv2 SA lifetimes can be configured by a Security Administrator based on [**selection**:*
  - *number of bytes,*
  - *length of time, where the time values can be configured within [**assignment**: integer range including 24] hours*

  *]*

].

**Application Note:** This element is unchanged from its definition in the Base-PP.

FCS_IPSEC_EXT.1.8

The TSF shall ensure that [**selection**:

- *IKEv1 Phase 2 SA lifetimes can be configured by a Security Administrator based on [**selection**:*
  - *number of bytes,*
  - *length of time, where the time values can be configured within [**assignment**: integer range including 8] hours*

  *],*
- *IKEv2 Child SA lifetimes can be configured by a Security Administrator based on [**selection**:*
  - *number of bytes,*
  - *length of time, where the time values can be configured within [**assignment**: integer range including 8] hours*

  *]*

].

**Application Note:** This element is unchanged from its definition in the Base-PP.

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 [**assignment**: *(one or more) numbers of bits that is at least twice the security strength of the negotiated DH group*] bits.

**Application Note:** This element is unchanged from its definition in the Base-PP.

FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in [**selection**: *IKEv1*, *IKEv2*] exchanges of length [**selection**:

- *according to the security strength associated with the negotiated DH group*,
- *at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash*

].

**Application Note:** This element is unchanged from its definition in the Base-PP.

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) according to RFC 5114 and**

[**selection**:

- *[**selection**: 14 (2048-bit MODP), 15 (3072-bit MODP), 16 (4096-bit MODP), 17 (6144-bit MODP), 18 (8192-bit MODP)] according to RFC 3526*,
- *[**selection**: 21 (521-bit Random ECP), 24 (2048-bit MODP with 256-bit POS, **no other DH Groups**] according to RFC 5114*

].

**Application Note:** This element has been modified from its definition in the NDcPP by mandating DH groups 19 and 20, both of which are selectable in the original definition of the element. Any groups other than 19 and 20 may be selected by the ST author but they are not required for conformance to this PP-Module.

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 [**selection**: *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 [**selection**: *IKEv1 Phase 2*, *IKEv2 CHILD_SA*] connection.

**Application Note:** This element is unchanged from its definition in the Base-PP.

FCS_IPSEC_EXT.1.13

The TSF shall ensure that [**selection**: *IKEv1*, *IKEv2*] protocols perform peer authentication using [**selection**: *RSA*, *ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [**selection**: *Pre-shared keys that conform to RFC 8784*, ***Pre-shared Keys transmitted via EAP-TTLS***, ***EAP-TLS***, *no other method*].

**Application Note:** At least one public-key-based Peer Authentication method is required in order to conform to this PP-Module; one or more of the public key schemes is chosen by the ST author to reflect what is implemented. The ST author also ensures that appropriate FCS requirements reflecting the algorithms used (and key generation capabilities, if provided) are listed to support those methods. Note that the TSS will elaborate on the way in which these algorithms are to be used.
If "IKEv1" is selected, "no other method" must be chosen in the third selection.
If a selection with "EAP-TLS" or "EAP-TTLS" is chosen, the selection-based requirement FCS_EAP_EXT.1 must be claimed. When an EAP method is used, verification occurs via an external authentication server. Though EAP-TTLS involves PSKs, FIA_PSK_EXT.1 is not claimed because PSK requirements for EAP-TTLS are addressed instead via the Authentication Server PP. EAP-based authentication is not supported for IKEv1.
If "pre-shared keys that conform to RFC 8784" is chosen, the selection-based requirement FIA_PSK_EXT.1 must be claimed. Since certificates are required for authentication, IKEv1 does not support PSKs, as PSK-based authentication and certificate-based authentication are mutually exclusive in IKEv1. Note that IKEv1 will be removed in a future version of this document.
Multifactor support can be achieved via traffic filtering in accordance with FPF_MFA_EXT.1.

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),** [**selection**: *SAN: IP address*, *SAN: Fully Qualified Domain Name (FQDN)*, *SAN: user FQDN*, *CN: IP address*, *CN: Fully Qualified Domain Name (FQDN)*, *CN: user FQDN*, ***no other reference identifier types***, ***[assignment: other supported reference identifier types]***].

**Application Note:** This PP-Module requires DN to be supported for certificate reference identifiers at minimum. Other selections may be made by the ST author but they are not required for conformance to this PP-Module.

#### 5.1.1.2 Identification and Authentication (FIA)

##### FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev

This is specified as a selection-based SFR in the Base-PP but is mandatory for any TOE that claims conformance to this PP-Module because a conformant TOE will always have the ability to receive an X.509 certificate from an external entity as part of IPsec communications. Therefore, a mechanism for the TSF to validate an X.509 certificate presented to it is required.

##### FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for **IPsec and [selection: *DTLS*, *HTTPS*, *SSH*, *TLS*, *no other protocols*]**, and [**selection**: *code signing for system software updates*, *[**assignment**: other uses], no additional uses*].

**Application Note:** The Base-PP allows the ST author to specify the TSF's use of X.509 certificates. Because this PP-Module mandates IPsec functionality, the SFR has been refined to force the inclusion of it. Other functions specified by the Base-PP may be chosen without restriction.

FIA_X509_EXT.2.2

When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [**selection, choose one of**: *allow the Administrator to choose whether to accept the certificate in these cases*, *accept the certificate*, *not accept the certificate*].

**Application Note:** This element is unchanged from its definition in the Base-PP.

##### FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1

This is specified as a selection-based SFR in the Base-PP but is mandatory for any TOE that claims conformance to this PP-Module because a conformant TOE will always have the ability to present an X.509 certificate to an external entity as part of IPsec communications. Therefore, a mechanism for the TSF to obtain a certificate for its own use is required.

#### 5.1.1.3 Security Management (FMT)

##### FMT_MTD.1/CryptoKeys Management of TSF Data

FMT_MTD.1.1/CryptoKeys

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

**Application Note:** This SFR, defined in the NDcPP as selection-based, is mandated for inclusion in this PP-Module because the refinements to FMT_SMF.1 mandate its inclusion. Note that it is also refined to refer specifically to keys and certificates used for VPN operation.

#### 5.1.1.4 Protection of the TSF (FPT)

##### FPT_TST_EXT.1 TSF Testing

FPT_TST_EXT.1.1

The TSF shall run a suite of the following self-tests [**selection**: *during initial start-up (on power on)*, *periodically during normal operation*, *at the request of the authorized user*, *at the conditions [**assignment**: conditions under which self-tests should occur]*] to demonstrate the correct operation of the TSF: **noise**

**source health tests,** [**assignment**: *list of self-tests run by the TSF*].

**Application Note:** This SFR is modified from its definition in the NDcPP by requiring noise source health tests to be performed regardless of what other testing is claimed. It is expected that the behavior of this testing will be described in the entropy documentation. Other self-tests may be defined at the ST author's discretion; note that the Application Note in the NDcPP regarding what other self-tests are expected is still applicable here.

**FPT_TUD_EXT.1 Trusted Update**

FPT_TUD_EXT.1.1

The TSF shall provide Security Administrators the ability to query the currently executing version of the TOE firmware/software and [**selection**: *the most recently installed version of the TOE firmware/software, no other TOE firmware/software version*].

**Application Note:** This element is unchanged from its definition in the Base-PP.

FPT_TUD_EXT.1.2

The TSF shall provide Security Administrators the ability to manually initiate updates to TOE firmware/software and [**selection**: *support automatic checking for updates, support automatic updates, no other update mechanism*].

**Application Note:** This element is unchanged from its definition in the Base-PP.

FPT_TUD_EXT.1.3

The TSF shall provide means to authenticate firmware/software updates to the TOE using a **digital signature mechanism and** [**selection**: *X.509 certificate, published hash*, ***no other mechanisms***] prior to installing those updates.

**Application Note:** The NDcPP provides an option for how firmware/software updates can be verified but this PP-Module requires the digital signature method to be selected at minimum. Note that all other options specified in the NDcPP for this component are permitted so it is possible for the TSF to use code signing certificates to validate updates, in which case FPT_TUD_EXT.2 from the Base-PP is also included in the ST.
If X.509 certificates are used to verify the integrity of an update, the certificates must conform to FIA_X509_EXT.1/Rev. Therefore, certificates that do not (or only partially) conform to FIA_X509_EXT.1/REV are not allowed as a means to authenticate firmware/software updates.
NDcPP states the ST author may use X.509 certificates that do not meet FIA_X509_EXT.1/Rev. This applies to trust anchors as they can be encoded as certificates. Even when they are encoded as certificates, the trust anchor must be protected by another mechanism that ensures its integrity and binds it to the 'code-signing' context. Trust anchors do not need to be validated according to FIA_X509_EXT.1, even if they are encoded as certificates; instead they need to be validated as trust anchors. FIA_X509_EXT.1/Rev does not require revocation checking of certificates designated as trust store elements. The integrity of trust store elements depends on administrative controls for loading and managing trust stores and functional integrity checks that are described in other SFRs.
So, if the certificate used to verify the update is a trust store element (self-signed and specifically trusted for verifying updates, with the integrity of this special purpose certificate protected by administrative controls and TOE integrity protections), then revocation checking is not required.
However, if the certificate is issued by a trusted root CA, or by a certificate authority which chains to a trusted root CA, then revocation checking is required for all elements of the certificate chain except the trusted root CA, and the TOE must be able to obtain fresh revocation information from an external source.

## 5.2 TOE Security Functional Requirements

The following section describes the SFRs that must be satisfied by any TOE that claims conformance to this PP-Module. These SFRs must be claimed regardless of which PP-Configuration is used to define the TOE.

### 5.2.1 Auditable Events for Mandatory SFRs

**Table 2: Auditable Events for Mandatory Requirements**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FAU_GEN.1/VPN | No events specified. | N/A |
| FCS_CKM.1/IKE | No events specified. | N/A |
| FMT_SMF.1/VPN | All administrative actions | No additional information |

| | | |
|---|---|---|
| FMT_SMF.1/VPN | All administrative actions | No additional information. |
| FPF_RUL_EXT.1 | Application of rules configured with the 'log' operation | Source and destination addresses<br>Source and destination ports<br>Transport layer protocol |
| FPT_FLS.1/SelfTest | No events specified. | N/A |
| FPT_TST_EXT.3 | No events specified. | N/A |
| FTP_ITC.1/VPN | Initiation of the trusted channel | No additional information. |
| FTP_ITC.1/VPN | Termination of the trusted channel | No additional information. |
| FTP_ITC.1/VPN | Failure of the trusted channel functions | Identification of the initiator and target of failed trusted channel establishment attempt |

### 5.2.2 Security Audit (FAU)

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

FAU_GEN.1.1/VPN

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

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

**Application Note:** The "Start-up and shutdown of the audit functions" event is identical to the event defined in the Base-PP's iteration of FAU_GEN.1. The TOE is not required to have two separate events for this behavior if there is only a single audit stream that which all audit events use. If the TOE does maintain a separate logging facility for VPN gateway-related behavior, then this event must be addressed for it. Note that if the audit functions cannot be started and stopped separately from the TOE itself, then auditing the start-up and shutdown of the TOE is sufficient to address this.

FAU_GEN.1.2/VPN

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

a. Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST, [*additional information defined in the Auditable Events for Mandatory Requirements table for each auditable event, where applicable*].

**Application Note:** The ST author only needs to include the auditable events that correspond to the SFRs claimed in the ST. The TOE is not required to generate auditable events for selection-based or optional SFRs that it does not claim.

### 5.2.3 Cryptographic Support (FCS)

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

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: **[selection:**

- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.3 for RSA schemes,***
- ***FIPS PUB 186-4, "Digital Signature Standard (DSS)", Appendix B.4 for ECDSA schemes and implementing "NIST curves" P-384 and [selection: P-256, P-521, no other curves]***

**] and [selection:**

- ***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 [selection: RFC 3526, RFC 7919],***

- ***no other key generation algorithm***

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

**Application Note:** The keys that are required to be generated by the TOE through this requirement are intended to be used for the authentication of the VPN peers during the IKE (either v1 or v2) key exchange. FCS_CKM.1 in the Base-PP is intended to be used for mechanisms required by the SFRs in the Base-PP. While it is required that the public key be associated with an identity in an X509v3 certificate, this association is not required to be performed by the TOE, and instead is expected to be performed by a CA in the OE.
As indicated in FCS_IPSEC_EXT.1, the TOE is required to implement RSA or ECDSA (or both) for peer authentication.
The generated key strength of 2048-bit RSA keys need to be equivalent to, or greater than, a symmetric key strength of 112 bits. See NIST Special Publication 800-57, “Recommendation for Key Management” for information about equivalent key strengths.

## 5.2.4 Security Management (FMT)

### FMT_SMF.1/VPN Specification of Management Functions

FMT_SMF.1.1/VPN

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

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

*[**selection**:*

- *Configuration of remote VPN client session timeout,*
- *Configuration of attributes used to deny establishment of remote VPN client sessions,*
- *Generation of bit-based pre-shared key,*
- *No other capabilities*

*]*].

**Application Note:** This SFR defines additional management functions for the TOE beyond what is defined in the Base-PP as FMT_SMF.1. The TOE may have all management functionality implemented in the same logical interface; it is not necessary for “network device management” and “VPN gateway management” to be implemented in separate interfaces.

## 5.2.5 Packet Filtering (FPF)

### FPF_RUL_EXT.1 Packet Filtering Rules

FPF_RUL_EXT.1.1

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

FPF_RUL_EXT.1.2

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

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

].

**Application Note:** This element identifies the protocols and references the protocol definitions that serve to define to what extent the network traffic can be interpreted by the TOE when importing (receiving network traffic or ingress)

and exporting (sending–or forming to be sent–network traffic or egress). While the protocol formatting specified in the RFCs is still used, many RFCs define behaviors which are no longer considered safe to follow. For example, RFC 792 defined the “Redirect” Internet Control Message Protocol (ICMP) type, which is not considered safe to honor when it might come from an adversary; the “source quench” message, which is insecure because its source cannot be validated.
It also identifies the various attributes that are applicable when constructing rules to be enforced by this requirement – the applicable interface is a property of the TOE and the rest of the identified attributes are defined in the associated RFCs. Note that the Protocol is the IPv4 field (in IPv6 this field is called the “next header”) that identifies the applicable protocol, such as TCP, UDP, ICMP, etc. Also, ‘Interface’ identified above is the external port where the applicable network traffic was received or alternately will be sent.

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.

**Application Note:** This element defines the operations that can be associated with rules used to match network traffic.

FPF_RUL_EXT.1.4

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

**Application Note:** This element identifies where rules can be assigned. Specifically, a conforming TOE must be able to assign filtering rules specific to each of its available and identifiable distinct network interfaces that handle layer 3 and 4 network traffic. Identifiable means the interface is unique and identifiable within the TOE, and does not necessarily require the interface to be visible from the network perspective (e.g., does not need to have an IP address assigned to it). A distinct network interface is one or more physical connections that share a common logical path into the TOE. For example, the TOE might have a small form-factor pluggable (SFP) port supporting SFP modules that expose a number of physical network ports, but since a common driver is used for all external ports they can be treated as a single distinct network interface. Note that there could be a separate ruleset for each interface or alternately a shared ruleset that somehow associates rules with specific interfaces.

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*].

**Application Note:** This element requires that an administrator is able to define the order in which configured filtering rules are processed for matches.

FPF_RUL_EXT.1.6

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

**Application Note:** This element requires that the behavior is always to deny network traffic when no rules apply.

### 5.2.6 Protection of the TSF (FPT)

#### FPT_FLS.1/SelfTest Failure with Preservation of Secure State (Self-Test Failures)

FPT_FLS.1.1/SelfTest

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

**Application Note:** This SFR defines the expected TSF response to failures of the self-tests defined in the Base-PP.

#### FPT_TST_EXT.3 Self-Test with Defined Methods

FPT_TST_EXT.3.1

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

FPT_TST_EXT.3.2

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

**Application Note:** This requirement expands upon the self-test requirements defined in the NDcPP by specifying the method by which one of the self-tests is to be performed. “Stored TSF executable code” refers to the entire software image of the device and not just the code related to the VPN gateway functionality defined by this PP-Module.

### 5.2.7 Trusted Path/Channels (FTP)

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

FTP_ITC.1.1/VPN

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

FTP_ITC.1.2/VPN

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

FTP_ITC.1.3/VPN

The TSF shall initiate communication via the trusted channel for [**selection, choose one of**: *remote VPN gateways or peers, no functions*].

**Application Note:** The FTP_ITC.1 requirement in the Base-PP relates to other trusted channel functions. This iteration is specific to IPsec VPN communications.

## 5.3 TOE Security Functional Requirements Rationale

The following rationale provides justification for each security objective for the TOE, showing that the SFRs are suitable to meet and achieve the security objectives:

**Table 3: SFR Rationale**

| Objective | Addressed by | Rationale |
|---|---|---|
| O.ADDRESS_FILTERING | FPF_RUL_EXT.1 | This SFR supports the objective by requiring the TSF to filter network traffic based on network address information. |
| | FTA_VCM_EXT.1 (optional) | This SFR supports the objective by optionally allowing the TOE to assign a private IP address to a VPN client so that traffic bound for an alternative address can be flagged as invalid. |
| O.AUTHENTICATION | FCS_IPSEC_EXT.1 (refined from Base-PP) | This SFR supports the objective by requiring the TOE's implementation of IPsec to include requirements for how the VPN client is authenticated. |
| | FIA_X509_EXT.1/Rev (from Base-PP) | This SFR supports the objective by requiring the TOE to implement X.509 validation functions so that it can authenticate remote entities that assert their identity using X.509 certificates. |
| | FIA_X509_EXT.2 (refined from Base-PP) | This SFR supports the objective by requiring the TOE to implement X.509 authentication functions so that it can authenticate remote entities that assert their identity using X.509 certificates. |
| | FIA_X509_EXT.3 (from Base-PP) | This SFR supports the objective by requiring the TOE to have the ability to generate a certificate request so that it can be issued an X.509 certificate that allows the TSF to offer proof of its own authenticity to external entities. |
| | FTP_ITC.1/VPN | This SFR supports the objective by requiring the TOE to use an IPsec trusted channel to communicate with external entities so that these entities may be authenticated. |
| | FTA_SSL.3/VPN (optional) | This SFR supports the objective by optionally allowing the TSF to terminate inactive VPN sessions so that an unattended session cannot be used to bypass authentication mechanisms. |

| | | |
|---|---|---|
| | FTA_TSE.1 (optional) | This SFR supports the objective by optionally defining alternative mechanisms to determine the validity of a subject to reject unauthorized or impersonated authentication attempts |
| | FPF_MFA_EXT.1 (optional) | This SFR supports the objective by optionally enforcing a multifactor authentication requirement on an IPsec connection. |
| | FCS_EAP_EXT.1 (selection-based) | This SFR supports the objective by optionally using an exteral authentication server to facilitate EAP-TLS or EAP-TTLS. |
| | FIA_HOTP_EXT.1 (selection-based) | This SFR supports the objective by optionally defining the implementation of HOTP as an authentication mechanism. |
| | FIA_PSK_EXT.1 (selection-based) | This SFR supports the objective by optionally supporting the use of pre-shared keys. |
| | FIA_PSK_EXT.2 (selection-based) | This SFR supports the objective by optionally specifying whether the TOE generates its own pre-shared keys or accepts them from an external source. |
| | FIA_PSK_EXT.3 (selection-based) | This SFR supports the objective by optionally defining the composition and use of password-based pre-shared keys used for authentication. |
| | FIA_TOTP_EXT.1 (selection-based) | This SFR supports the objective by optionally defining the implementation of TOTP as an authentication mechanism. |
| O.CRYPTOGRAPHIC_FUNCTIONS | FCS_COP.1/DataEncryption (refined from Base-PP) | This SFR supports the objective by requiring the TOE to implement AES in a specified manner. |
| | FCS_IPSEC_EXT.1 (refined from Base-PP) | This SFR supports the objective by requiring the TOE to implement the IPsec protocol in a specified manner |
| | FCS_CKM.1/IKE | This SFR supports the objective by requiring the TOE to generate cryptographic keys used for Internet Key Exchange (IKE) in a specified manner. |
| | FCS_EAP_EXT.1 (selection-based) | This SFR supports the objective by optionally using an MSK computed by an external authentication server via the EAP-TLS or EAP-TTLS method. |
| | FIA_PSK_EXT.1 (selection-based) | This SFR supports the objective by supporting a mechanism by which PSKs can be used to fortify encryption. |
| | FIA_PSK_EXT.2 (selection-based) | This SFR supports the objective by optionally specifying whether the TOE generates its own pre-shared keys or accepts them from an external source.. |
| O.FAIL_SECURE | FPT_TST_EXT.1 (refined from Base-PP) | This SFR supports the objective by requiring the TOE to execute self-tests that allow the TSF to determine if it is in a failed state. |
| | FPT_TUD_EXT.1 (refined from Base-PP) | This SFR supports the objective by requiring the TOE to validate software updates before applying them to reduce the risk of the TOE entering a failed state. |
| | FPT_FLS.1/SelfTest | This SFR supports the objective by requiring the TOE to preserve a secure state if a self-test failure is detected. |
| | FPT_TST_EXT.3 | This SFR supports the objective by requiring the TOE to verify the integrity of its executable code to ensure that it will operate in a known state. |
| O.PORT_FILTERING | FPF_RUL_EXT.1 | This SFR supports the objective by requiring the TSF to filter network traffic based on port information. |
| O.SYSTEM_MONITORING | FAU_GEN.1/VPN | This SFR supports the objective by requiring the TOE to generate security-relevant audit events related to VPN gateway functionality. |

| | | |
|---|---|---|
| | FPF_RUL_EXT.1 | This SFR supports the objective by requiring the TOE to have the ability to log network traffic that matches certain characteristics. |
| O.TOE_ADMINISTRATION | FMT_MTD.1/CryptoKeys (refined from Base-PP) | This SFR supports the objective by requiring the TOE to implement a key management function and ensure that only authorized users can use it. |
| | FMT_SMF.1/VPN | This SFR supports the objective by specifying the management functions required specifically for VPN gateway functionality. |

## 5.4 TOE Security Assurance Requirements

This PP-Module does not define any SARs beyond those defined within the Base-PPs to which it can claim conformance. It is important to note that a TOE that is evaluated against this PP-Module is inherently evaluated against the NDcPP as well. This PP includes a number of EAs associated with both Security Functional Requirements (SFRs) and SARs. Additionally, this PP-Module includes a number of SFR-based EAs that similarly refine the SARs of the Base-PPs. The evaluation laboratory will evaluate the TOE against the chosen Base-PP and supplement that evaluation with the necessary SFRs that are taken from this PP-Module.

# 6 Consistency Rationale

## 6.1 collaborative Protection Profile for Network Devices

### 6.1.1 Consistency of TOE Type

When this PP-Module is used to extend the NDcPP, the TOE type for the overall TOE is still a network device. The TOE boundary is simply extended to include VPN gateway functionality that is provided by the network device.

### 6.1.2 Consistency of Security Problem Definition

The threats, assumptions, and organizational security policies (OSPs) defined by this PP-Module (see sections 3.1 through 3.3) supplement those defined in the NDcPP as follows:

| PP-Module Threat, Assumption, OSP | Consistency Rationale |
|---|---|
| T.DATA_INTEGRITY | The threat of data integrity compromise is a specific example of the T.WEAK_CRYPTOGRAPHY threat defined in the Base-PP. |
| T.NETWORK_ACCESS | The threat of a malicious entity accessing protected network resources without authorization is a specific example of the T.UNTRUSTED_COMMUNICATION_CHANNELS threat defined in the Base-PP. |
| T.NETWORK_DISCLOSURE | Exposure of network devices due to insufficient protection is a specific example of the T.UNTRUSTED_COMMUNICATION_CHANNELS threat defined in the Base-PP. |
| T.NETWORK_MISUSE | Depending on the specific nature of the misuse of network resources, this threat is a specific manifestation of either the T.UNTRUSTED_COMMUNICATION_CHANNELS or T.WEAK_AUTHENTICATION_ENDPOINTS threat defined in the Base-PP. |
| T.REPLAY_ATTACK | A replay attack is mentioned in the Base-PP as a specific type of attack based on the T.UNTRUSTED_COMMUNICATION_CHANNELS threat. |
| A.CONNECTIONS | This assumption defines the TOE’s placement in a network such that it is able to perform its required security functionality. The Base-PP does not define any assumptions about the TOE’s architectural deployment so there is no conflict here. |

### 6.1.3 Consistency of Objectives

The objectives for the TOEs are consistent with the NDcPP based on the following rationale:

| PP-Module TOE Objective | Consistency Rationale |
|---|---|
| O.ADDRESS_FILTERING | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.AUTHENTICATION | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.CRYPTOGRAPHIC_FUNCTIONS | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.FAIL_SECURE | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.PORT_FILTERING | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.SYSTEM_MONITORING | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |
| O.TOE_ADMINISTRATION | The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it. |

The objectives for the TOE's OE are consistent with the NDcPP based on the following rationale:

| PP-Module OE Objective | Consistency Rationale |
|---|---|

| | |
|---|---|
| OE.CONNECTIONS | This objective intends for the TOE to be connected to environmental networks in such a way that its primary functionality can be appropriately enforced. There is no inconsistency here with respect to the Base-PP because the Base-PP does not define any restrictions on how a network device is connected to its environment. |

### 6.1.4 Consistency of Requirements

This PP-Module identifies several SFRs from the NDcPP that are needed to support functionality. This is considered to be consistent because the functionality provided by the NDcPP is being used for its intended purpose. The PP-Module also identifies a number of modified SFRs from the NDcPP that are used entirely to provide functionality for . The rationale for why this does not conflict with the claims defined by the NDcPP are as follows:

| PP-Module Requirement | Consistency Rationale |
|---|---|
| **Modified SFRs** | |
| FCS_COP.1/DataEncryption | This PP-Module restricts the Base-PP SFR to a subset of existing permissible functionality and does not introduce any new behavior. |
| FCS_IPSEC_EXT.1 | This PP-Module restricts the Base-PP SFR to a subset of existing permissible functionality and does not introduce any new behavior. |
| FIA_X509_EXT.1/Rev | This PP-Module does not modify the Base-PP SFR; it only mandates the inclusion of the SFR because a conformant TOE will always require this functionality that is only conditional in the Base-PP. |
| FIA_X509_EXT.2 | This PP-Module restricts the Base-PP SFR to a subset of existing permissible functionality and does not introduce any new behavior. |
| FIA_X509_EXT.3 | This PP-Module does not modify the Base-PP SFR; it only mandates the inclusion of the SFR because a conformant TOE will always require this functionality that is only conditional in the Base-PP. |
| FMT_MTD.1/CryptoKeys | This PP-Module applies the key management functionality already defined in the Base-PP specifically to functionality related to VPN gateways. |
| FPT_TST_EXT.1 | This PP-Module refines the Base-PP SFR to mandate a specific type of self-test. This is consistent with the Base-PP because the execution of this self-test is already implied by the Base-PP through its entropy requirements. |
| FPT_TUD_EXT.1 | This PP-Module restricts the Base-PP SFR to a subset of existing permissible functionality and does not introduce any new behavior. |
| **Additional SFRs** | |
| This PP-Module does not add any requirements when the NDcPP is the base. | |
| **Mandatory SFRs** | |
| FAU_GEN.1/VPN | This SFR adds new auditable events for the TOE that relate to the functionality that is introduced by the PP-Module. |
| FCS_CKM.1/IKE | This PP-Module specifies a method of key generation that is not defined in the Base-PP. This is used for functionality defined in the Base-PP (IKE) that this PP-Module chooses to represent in greater detail. |
| FMT_SMF.1/VPN | This SFR defines management functions that are specific to the functionality required by this PP-Module and were therefore not already defined in the Base-PP iteration of it. |
| FPF_RUL_EXT.1 | This SFR defines specific behavior for the processing of network traffic, specifically which communications channel is used based on certain attributes of the traffic. The Base-PP does not apply any constraints on how usage of a trusted channel is controlled so this does not contradict anything presented in the Base-PP. |
| FPT_FLS.1/SelfTest | The Base-PP already requires the TOE to specify the self-tests that are performed. This PP-Module simply goes one step further and requires the TSF to behave in a certain way upon failure of those self-tests. |
| FPT_TST_EXT.3 | This PP-Module adds to the self-testing requirements from the Base-PP by mandating that a specific self-test be performed and that it be performed in a certain manner. This does not conflict with the Base-PP because the method used to perform the self-test is a cryptographic function already mandated by the Base-PP. |
| | |

| | |
|---|---|
| FTP_ITC.1/VPN | This PP-Module iterates a Base-PP SFR to refer to an interface that is unique to the PP-Module. This does not affect the ability of the Base-PP iteration of the SFR to be satisfied. |
| **Optional SFRs** | |
| FPF_MFA_EXT.1 | This SFR relates specifically to the handling of traffic that is used for the establishment of IPsec connections. |
| **Selection-based SFRs** | |
| FCS_EAP_EXT.1 | This SFR defines the use of EAP-TLS; the Base-PP already defines requirements for TLS so potential support for EAP-TLS is consistent with functionality that the Base-PP already expects the TOE may have. |
| FIA_HOTP_EXT.1 | This SFR relates to use of pre-shared keys, which is behavior that only applies to the establishment of IPsec connections. |
| FIA_PSK_EXT.1 | This SFR defines the use of pre-shared keys, which is behavior that only relates to the establishment of IPsec connections. |
| FIA_PSK_EXT.2 | This SFR relates to use of pre-shared keys, which is behavior that only applies to the establishment of IPsec connections. |
| FIA_PSK_EXT.3 | This SFR relates to use of pre-shared keys, which is behavior that only applies to the establishment of IPsec connections. |
| FIA_TOTP_EXT.1 | This SFR relates to use of pre-shared keys, which is behavior that only applies to the establishment of IPsec connections. |
| **Objective SFRs** | |
| This PP-Module does not define any Objective requirements. | |
| **Implementation-based SFRs** | |
| FTA_SSL.3/VPN | This SFR refers to a specific condition under which a trusted channel is terminated by the TSF. The Base-PP supports termination of trusted channels and does not mandate this be done in any particular method. |
| FTA_TSE.1 | This SFR refers to a specific condition under which a trusted channel is terminated by the TSF. The Base-PP supports termination of trusted channels and does not mandate this be done in any particular method. |
| FTA_VCM_EXT.1 | This SFR refers to network addressing, which is outside the scope of the Base-PP and therefore not prohibited by it. |

# Appendix A - Optional SFRs

## A.1 Strictly Optional Requirements

### A.1.1 Auditable Events for Strictly Optional SFRs

**Table 4: Auditable Events for Optional Requirements**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FPF_MFA_EXT.1 | No events specified. | N/A |

### A.1.2 Packet Filtering (FPF)

The TOE may support multifactor authentication by blocking all other traffic after connection is established until secondary authentication is validated.

#### FPF_MFA_EXT.1 Multifactor Authentication Filtering

FPF_MFA_EXT.1.1

The TSF shall not forward packets to the internal network until the IKE tunnel has been established, except those necessary to verify additional authentication factors of the client.

FPF_MFA_EXT.1.2

The TSF shall [**selection**: *verify*, *verify via an external authentication server*] additional authentication factors of the client.

**Application Note:** If "verify" is selected, FIA_PSK_EXT.1 must be claimed.

## A.2 Objective Requirements

This PP-Module does not define any Objective SFRs.

## A.3 Implementation-based Requirements

### A.3.1 Auditable Events for Implementation-Dependent SFRs

**Table 5: Auditable Events for Feat-based Requirements**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FTA_SSL.3/VPN | No events specified. | N/A |
| FTA_TSE.1 | No events specified. | N/A |
| FTA_VCM_EXT.1 | No events specified. | N/A |

### A.3.2 TOE Access (FTA)

This section contains requirements that may be optionally selected by the ST author for a “headend” VPN gateway device. The requirements in the main body of this PP-Module are those determined necessary for a multi-site VPN gateway appliance. Another application of a VPN appliance is in an architecture that is intended to serve mobile users, by providing a secure means in which a remote client may access a trusted network. These devices provide the capability to manage remote VPN clients (e.g., assigning IP addresses, managing client sessions) that are not necessarily found in VPN gateways that are limited to providing a secure communication path between trusted networks. Rather than mandate that all VPN gateways provide this mobility aspect, the requirements below are specified as an option. What this means is that multi-site VPN gateways do not have to provide these capabilities, but those devices wishing to serve the mobility community should implement the optional requirements from this Appendix in addition to all mandatory and selection-based requirements that apply to them.

#### FTA_SSL.3/VPN TSF-Initiated Termination (VPN Headend)

FTA_SSL.3.1/VPN

The TSF shall terminate **a remote VPN client** session after [*an Administrator-configurable time interval of session inactivity*].

**Application Note:** This requirement exists in the NDcPP; however, it is intended to address a remote administrative interactive session. Here, the requirement applies to a VPN client that has established an SA. After some configurable time period without any activity, the connection between the VPN

headend and client is terminated.

### FTA_TSE.1 TOE Session Establishment

FTA_TSE.1.1

The TSF shall be able to deny session establishment **of a remote VPN client session** based on [*location, time, day, [**selection**: no other attributes, [**assignment**: other attributes]]*].

**Application Note:** For this PP-Module, “location” is defined as the client’s IP address.

### FTA_VCM_EXT.1 VPN Client Management

FTA_VCM_EXT.1.1

The TSF shall assign a private IP address to a VPN client upon successful establishment of a security session.

**Application Note:** For this requirement, the private IP address is one that is internal to the trusted network for which the TOE is the headend.

# Appendix B - Selection-based Requirements

## B.1 Auditable Events for Selection-based SFRs

**Table 6: Auditable Events for Selection-based Requirements**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FCS_EAP_EXT.1 | No events specified. | N/A |
| FIA_HOTP_EXT.1 | No events specified. | N/A |
| FIA_PSK_EXT.1 | No events specified. | N/A |
| FIA_PSK_EXT.2 | No events specified. | N/A |
| FIA_PSK_EXT.3 | No events specified. | N/A |
| FIA_TOTP_EXT.1 | No events specified. | N/A |

## B.2 Cryptographic Support (FCS)

### FCS_EAP_EXT.1 EAP-TLS

***The inclusion of this selection-based component depends upon selection in FCS_IPSEC_EXT.1.13.***

FCS_EAP_EXT.1.1

The TSF shall implement [**selection**: *EAP-TLS protocol as specified in RFC 5216 and updated by RFC 8996*, *EAP-TTLS as specified in RFC 5281*] over a protected channel per FTP_ITC.1 **from the Base-PP** with an authentication server.

FCS_EAP_EXT.1.2

The TSF shall implement EAP-TLS or EAP-TTLS with the TSF as the EAP client, an authentication server as the EAP server, and the VPN peer as the supplicant.

FCS_EAP_EXT.1.3

The TSF shall use the MSK from the [**selection**: *EAP-TLS*, *EAP-TTLS*] response as the IKEv2 shared secret in the authentication payload.

**Application Note:** This SFR covers the EAP requirements for a client connection to an authentication server. In particular, authentication server functionality implemented in the same device, rather than in an external device, is covered under the Authentication Server PP-Module, and is out of scope for this PP-Module. The MSK derived from EAP is distinct from the PSK. The MSK is substituted in for the IKEv2 shared secret in the AUTH computations.

## B.3 Identification and Authentication (FIA)

The TOE may support pre-shared keys for use in the IPsec protocol, and may use pre-shared keys in other protocols as well. PSK in the context of this document refers to generated values, memorized values subject to conditioning, one-time passwords, and combinations as described in FIA_PSK_EXT.1.2.

### FIA_HOTP_EXT.1 HMAC-Based One-Time Password Pre-Shared Keys

FIA_HOTP_EXT.1.1

The TSF shall support HMAC-Based One-Time Password (HOTP) authentication in accordance with RFC 4226 to authenticate the user before establishing VPN connection.

FIA_HOTP_EXT.1.2

The TSF shall generate an HOTP seed according to FCS_RBG_EXT.1 of [**selection**: *128*, *256*] bits.

FIA_HOTP_EXT.1.3

The TSF shall generate a new HOTP seed value for each client.

FIA_HOTP_EXT.1.4

The TSF shall use [**selection**: *SHA-1*, *SHA-256*, *SHA-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *160*, *256*, *384*, *512*] to derive an HOTP hash from the HOTP seed and counter.

FIA_HOTP_EXT.1.5

The TSF shall truncate the HOTP hash per FIA_HOTP_EXT.1.4 to create an HOTP of [**selection**:

- *administrator configurable character length of at least 6*,
- *preset character length of [**selection, choose one of**: 6, 7, 8, 9, 10]*

].

FIA_HOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**: administrator configurable value, [**assignment**: value less than 10]] per minute*,
- *lock the associated account after [**selection**: administrator configurable value, [**assignment**: value less than 10]] failed attempts until [**selection**: an administrator unlocks the account, a configurable time period]*

].

FIA_HOTP_EXT.1.7

The TSF shall not verify HOTP attempts outside of the counter look ahead window of [**selection**: *a configurable value*, [**assignment**: *a value less than or equal to 3]*] for resynchronization.

FIA_HOTP_EXT.1.8

The TSF shall increment the counter after each successful authentication.

**Application Note:** The selection FIA_HOTP_EXT.1.4 must be consistent with the key size specified for the size of the keys used in conjunction with the keyed-hash message authentication.
In FIA_HOTP_EXT.1.5 the ST author may either provide a configurable character length of at least 6 or a preset size between 6 and 10.
In FIA_HOTP_EXT.1.6 the ST author may select throttle requests, account lockout, or both.
The HOTP seed and all derived values are considered secret keys for purposes of protection.

### FIA_PSK_EXT.1 Pre-Shared Key Composition

***The inclusion of this selection-based component depends upon selection in FCS_IPSEC_EXT.1.13, FPF_MFA_EXT.1.2.***

FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for IPsec and [**selection**: *IKEv2*, *multifactor authentication filtering*].

**Application Note:** If “IKEv2” is selected, then the corresponding IKEv2 selection must be made in FCS_IPSEC_EXT.1.13. If “multifactor authentication filtering” is selected, then “verify” should be selected in FPF_MFA_EXT.1.2.

FIA_PSK_EXT.1.2

The TSF shall be able to accept the following as pre-shared keys: [**selection**: *generated bit-based*, *password-based*, *HMAC-based one-time password*, *time-based one-time password*, *combination of a generated bit-based and HMAC-based one-time password*, *combination of a generated bit-based and time-based one-time password*, *combination of a password-based and HMAC-based one-time password*, *Combination of a password-based and time-based one-time password*] keys.

**Application Note:** This requirement is included if “pre-shared keys that conform to RFC 8784” is selected in FCS_IPSEC_EXT.1.13, or if “verify” is selected in FPF_MFA_EXT.1.2. If these selections are made in both FCS_IPSEC_EXT.1.13 and FPF_MFA_EXT.1.2, the TSF must satisfy the appropriate FIA_PSK_EXT.1 selections for each.
If any selection including "generated bit-based" is chosen, then FIA_PSK_EXT.2 must be included. If “pre-shared keys that conform to RFC 8784” is selected in FCS_IPSEC_EXT.1.13, a generated, bit-based PSK must be used.
If any selection including Password-based keys is chosen, then FIA_PSK_EXT.3 must be included.
If any selection including HMAC-based one-time password keys is chosen, then FIA_HOTP_EXT.1 must be included.

If any selection including time-based one-time password is chosen, then FIA_TOTP_EXT.1 must be included.

The first four selections are for single factor authentication options, the last four selections are for multifactor authentication options.

### FIA_PSK_EXT.2 Generated Pre-Shared Keys

***The inclusion of this selection-based component depends upon selection in FIA_PSK_EXT.1.2.***

FIA_PSK_EXT.2.1

The TSF shall be able to [**selection**:

- *accept externally generated pre-shared keys,*
- *generate [**selection**: 128, 256] bit-based pre-shared keys via FCS_RBG_EXT.1.*

]

**Application Note:** Generated PSKs are expected to be shared between components via an out of band mechanism.
This requirement is selection dependent on FIA_PSK_EXT.1.

### FIA_PSK_EXT.3 Password-Based Pre-Shared Keys

***The inclusion of this selection-based component depends upon selection in FIA_PSK_EXT.1.2.***

FIA_PSK_EXT.3.1

The TSF shall support a PSK of up to [**assignment**: *positive integer of 64 or more*] characters.

FIA_PSK_EXT.3.2

The TSF shall allow PSKs to be composed of any combination of upper case characters, lower case characters, numbers, and the following special characters: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")", and [**selection**: *[**assignment**: other supported special characters], no other characters*]

FIA_PSK_EXT.3.3

The TSF shall perform Password-based Key Derivation Functions in accordance with a specified cryptographic algorithm HMAC- [**selection**: *SHA-256*, *SHA-384*, *SHA-512*], with [**assignment**: *positive integer of 4096 or more*] iterations, and output cryptographic key sizes [**selection**: *128*, *256*] that meet the following: [NIST SP 800-132].

FIA_PSK_EXT.3.4

The TSF shall not accept PSKs less than [**selection**: *a value settable by the administrator, [**assignment**: minimum PSK length accepted by the TOE, must be >= 6]*] and greater than the maximum PSK length defined in FIA_PSK_EXT.3.1.

FIA_PSK_EXT.3.5

The TSF shall generate all salts using an RBG that meets FCS_RBG_EXT.1 and with entropy of [**assignment**: *value equal to or greater than 128*] bits.

FIA_PSK_EXT.3.6

The TSF shall require the PSK to be entered before every initiated connection.

FIA_PSK_EXT.3.7

The TSF shall [**selection**: *provide a password strength meter, check the password against a denylist, perform no action to assist the user in choosing a strong password*].

**Application Note:** For FIA_PSK_EXT.3.1, the ST author assigns the maximum size of the PSK it supports; it must support at least 64 characters or a length defined by the platform.
For FIA_PSK_EXT.3.2, the ST author assigns any other supported characters; if there are no other supported characters, they should select "no other characters."
For FIA_PSK_EXT.3.3, the ST author selects the parameters based on the PBKDF used by the TSF.
For FIA_PSK_EXT.3.4 If the minimum length is settable, then the ST author chooses "a value settable by the administrator." If the minimum length is not settable, the ST author fills in the assignment with the minimum length the PSK

must be. This requirement is to ensure bounds work properly.
For FIA_PSK_EXT.3.7, the ST author may select one, both, or neither of the functions in alignment with NIST SP 800-63b.
This requirement is selection dependent on FIA_PSK_EXT.1.

### FIA_TOTP_EXT.1 Time-Based One-Time Password Pre-Shared Keys

FIA_TOTP_EXT.1.1

The TSF shall support Time-Based One-Time Password (TOTP) authentication in accordance with RFC 6238 to authenticate the user before establishing VPN connection.

FIA_TOTP_EXT.1.2

The TSF shall generate a TOTP seed according to FCS_RBG_EXT.1 of [**selection**: *128*, *256*] bits.

FIA_TOTP_EXT.1.3

The TSF shall generate a new TOTP seed for each client.

FIA_TOTP_EXT.1.4

The TSF shall use [**selection**: *SHA-1*, *SHA-256*, *SHA-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *160*, *256*, *384*, *512*] to derive a TOTP hash from the TOTP seed and current time provided by NTP.

FIA_TOTP_EXT.1.5

The TSF shall truncate the TOTP hash per FIA_TOTP_EXT.1.4 to create a TOTP of [**selection**:

- *administrator configurable character length of at least 6*,
- *preset character length of [**selection, choose one of**: 6, 7, 8, 9, 10]*

].

FIA_TOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**: administrator configurable value, [**assignment**: value less than 10]] per minute*,
- *lock the associated account after [**selection**: administrator configurable value, [**assignment**: value less than 10]] failed attempts until [**selection**: an administrator unlocks the account, a configurable time period]*

].

FIA_TOTP_EXT.1.7

The TSF shall set a time-step size of [**selection**: *a configurable value*, *[**assignment**: a value less than or equal to 30]*] seconds.

FIA_TOTP_EXT.1.8

The TSF shall not validate a drift of more than [**selection**: *a configurable value*, *[**assignment**: a value less than or equal to 3]*] time-steps.

FIA_TOTP_EXT.1.9

The TSF shall [**selection**: *allow resynchronization by recording time drift within the limit of FIA_TOTP_EXT.1.8*, *not permit resynchronization*].

**Application Note:** The selection FIA_TOTP_EXT.1.4 must be consistent with the key size specified for the size of the keys used in conjunction with the keyed-hash message authentication.
In FIA_TOTP_EXT.1.5 the ST author may either provide a configurable character length of at least 6 or a preset size between 6 and 10.
In FIA_TOTP_EXT.1.6 the ST author may select throttle requests, account lockout, or both.
The TOTP seed and all derived values are considered secret keys for purposes of protection.

# Appendix C - Extended Component Definitions

This appendix contains the definitions for all extended requirements specified in the Module.

## C.1 Extended Components Table

All extended components specified in the Module are listed in this table:

**Table 7: Extended Component Definitions**

| Functional Class | Functional Components |
|---|---|
| Packet Filtering (FPF) | FPF_MFA_EXT Multifactor Authentication Filtering<br>FPF_RUL_EXT Packet Filtering Rules |
| Protection of the TSF (FPT) | FPT_TST_EXT TSF Self-Test |
| Cryptographic Support (FCS) | FCS_EAP_EXT EAP-TLS |
| Identification and Authentication (FIA) | FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys<br>FIA_PSK_EXT Pre-Shared Key Composition<br>FIA_TOTP_EXT Time-Based One-Time Password Pre-Shared Keys |
| TOE Access (FTA) | FTA_VCM_EXT VPN Client Management |

## C.2 Extended Component Definitions

### C.2.1 Packet Filtering (FPF)

This class contains families that describe packet filtering behavior. Packet filtering refers to the notion that network traffic that is transmitted “through” the TOE (i.e. the source and destination of the traffic is not the TOE but the TOE is on the routing path between these two entities) can be treated differently by the TSF based on attributes associated with the traffic. As this class is defined solely to contain an extended component defined for this PP-Module, it has two families, FPF_MFA_EXT and FPF_RUL_EXT.

#### C.2.1.1 FPF_RUL_EXT Packet Filtering Rules

**Family Behavior**

This family defines the requirements for the rules that are used to perform packet filtering of network traffic.

**Component Leveling**

FPF_RUL_EXT ——— 1

FPF_RUL_EXT.1, Packet Filtering Rules, requires the TSF to enforce a given set of packet filtering rules in an administrator-defined order against one or more TOE interfaces.

**Management: FPF_RUL_EXT.1**

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

- Ability to configure the TOE’s packet filtering functionality (i.e. the operations to be performed on network traffic based on configured attributes, the interfaces that these are associated with, and the order in which they are applied).

**Audit: FPF_RUL_EXT.1**

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

- Application of rules configured with the ‘log’ operation (including source and destination address, source and destination port, and transport layer protocol value).

**FPF_RUL_EXT.1 Packet Filtering Rules**

Hierarchical to: No other components.

Dependencies to: No dependencies.

**FPF_RUL_EXT.1.1**

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

#### FPF_RUL_EXT.1.2

The TSF shall allow the definition of packet filtering rules using the following network protocols and protocol fields: [**assignment**: *supported network protocols and protocol fields*].

#### FPF_RUL_EXT.1.3

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

#### FPF_RUL_EXT.1.4

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

#### FPF_RUL_EXT.1.5

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

#### FPF_RUL_EXT.1.6

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

### C.2.1.2 FPF_MFA_EXT Multifactor Authentication Filtering

#### Family Behavior

Components in this family describe the requirements for multifactor authentication filtering when using the VPN client.

#### Component Leveling

FPF_MFA_EXT ——— 1

FPF_MFA_EXT.1, Multifactor Authentication Filtering, defines the use and composition of multifactor authentication filtering.

#### Management: FPF_MFA_EXT.1

No specific management functions are identified.

#### Audit: FPF_MFA_EXT.1

No specific audit functions are identified.

#### FPF_MFA_EXT.1 Multifactor Authentication Filtering

Hierarchical to: No other components.

Dependencies to: No dependencies.

#### FPF_MFA_EXT.1.1

The TSF shall not forward packets to the internal network until the IKE tunnel has been established, except those necessary to verify additional authentication factors of the client.

#### FPF_MFA_EXT.1.2

The TSF shall [**selection**: *verify, verify via an external authentication server*] additional authentication factors of the client.

## C.2.2 Protection of the TSF (FPT)

This Module defines the following extended components as part of the FPT class originally defined by CC Part 2:

### C.2.2.1 FPT_TST_EXT TSF Self-Test

#### Family Behavior

This family is defined in the Base-PP. This PP-Module augments the extended family by adding one additional component, FPT_TST_EXT.3. This new component and its impact on the extended family's component leveling are shown below; reference the Base-PP for all other definitions for this family.

#### Component Leveling

FPT_TST_EXT ——— 3

FPT_TST_EXT.3, Self-Test with Defined Methods, requires the TSF to specify the methods by which self-

testing is performed in addition to identifying the self-tests that are executed and the circumstances in which this execution occurs.

#### Management: FPT_TST_EXT.3

No specific management functions are identified.

#### Audit: FPT_TST_EXT.3

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

- Indication that TSF self-test was completed
- Failure of self-test

#### FPT_TST_EXT.3 Self-Test with Defined Methods

Hierarchical to: No other components.

Dependencies to: No dependencies.

##### FPT_TST_EXT.3.1

The TSF shall run a suite of the following self-tests [**assignment**: *timing when self-testing is run*] to demonstrate the correct operation of the TSF: [**assignment**: *list of self-tests performed*].

##### FPT_TST_EXT.3.2

The TSF shall execute the self-testing through [**assignment**: *self-testing mechanism*].

### C.2.3 Cryptographic Support (FCS)

This Module defines the following extended components as part of the FCS class originally defined by CC Part 2:

### C.2.3.1 FCS_EAP_EXT EAP-TLS

#### Family Behavior

Components in this family describe the requirements for EAP-TLS.

#### Component Leveling

FCS_EAP_EXT ——— 1

FCS_EAP_EXT.1, EAP-TLS, defines the use of EAP-TLS.

#### Management: FCS_EAP_EXT.1

No specific management functions are identified.

#### Audit: FCS_EAP_EXT.1

No specific audit functions are identified.

#### FCS_EAP_EXT.1 EAP-TLS

Hierarchical to: No other components.

Dependencies to: FCS_IPSEC_EXT.1 IPsec Protocol

##### FCS_EAP_EXT.1.1

The TSF shall implement [**selection**: *EAP-TLS protocol as specified in RFC 5216 and updated by RFC 8996*, *EAP-TTLS as specified in RFC 5281*] over a protected channel per FTP_ITC.1 with an authentication server.

##### FCS_EAP_EXT.1.2

The TSF shall implement EAP-TLS or EAP-TTLS with the TSF as the EAP client, an authentication server as the EAP server, and the VPN peer as the supplicant.

##### FCS_EAP_EXT.1.3

The TSF shall use the MSK from the [**selection**: *EAP-TLS*, *EAP-TTLS*] response as the IKEv2 shared secret in the authentication payload.

### C.2.4 Identification and Authentication (FIA)

This Module defines the following extended components as part of the FIA class originally defined by CC Part

2:

## C.2.4.1 FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys

### Family Behavior

Components in this family define requirements for the use of HMAC-Based One-Time password authentication, including generation methods and usage restrictions.

### Component Leveling

FIA_HOTP_EXT ——— 1

FIA_HOTP_EXT.1, HMAC-Based One-Time Password Pre-Shared Keys, defines the implementation of HOTP.

### Management: FIA_HOTP_EXT.1

No specific management functions are identified.

### Audit: FIA_HOTP_EXT.1

No specific audit functions are identified.

### FIA_HOTP_EXT.1 HMAC-Based One-Time Password Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FCS_COP.1 Cryptographic Operation

#### FIA_HOTP_EXT.1.1

The TSF shall support HMAC-Based One-Time Password (HOTP) authentication in accordance with RFC 4226 to authenticate the user before establishing VPN connection.

#### FIA_HOTP_EXT.1.2

The TSF shall generate an HOTP seed according to FCS_RBG_EXT.1 of [**selection**: *128*, *256*] bits.

#### FIA_HOTP_EXT.1.3

The TSF shall generate a new HOTP seed value for each client.

#### FIA_HOTP_EXT.1.4

The TSF shall use [**selection**: *SHA-1*, *SHA-256*, *SHA-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *160*, *256*, *384*, *512*] to derive an HOTP hash from the HOTP seed and counter.

#### FIA_HOTP_EXT.1.5

The TSF shall truncate the HOTP hash per FIA_HOTP_EXT.1.4 to create an HOTP of [**selection**:

- *administrator configurable character length of at least 6*,
- *preset character length of [**selection, choose one of**: 6, 7, 8, 9, 10]*

].

#### FIA_HOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**: administrator configurable value, [**assignment**: value less than 10]] per minute*,
- *lock the associated account after [**selection**: administrator configurable value, [**assignment**: value less than 10]] failed attempts until [**selection**: an administrator unlocks the account, a configurable time period]*

].

#### FIA_HOTP_EXT.1.7

The TSF shall not verify HOTP attempts outside of the counter look ahead window of [**selection**: *a configurable value*, [**assignment**: *a value less than or equal to 3]*] for resynchronization.

#### FIA_HOTP_EXT.1.8

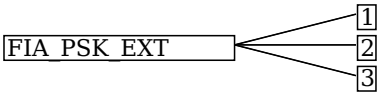
The TSF shall increment the counter after each successful authentication.

## C.2.4.2 FIA_PSK_EXT Pre-Shared Key Composition

### Family Behavior

This family defines requirements for what the TSF defines or generates as an acceptably strong pre-shared key for authentication.

### Component Leveling




FIA_PSK_EXT.1, Pre-Shared Key Composition, defines the use and composition of pre-shared keys used for IPsec.

FIA_PSK_EXT.2, Generated Pre-Shared Keys, defines the use and composition of generated pre-shared keys used for IPsec.

FIA_PSK_EXT.3, Password-Based Pre-Shared Keys, defines the use and composition of password-based pre-shared keys used for IPsec.

### Management: FIA_PSK_EXT.1

No specific management functions are identified.

### Audit: FIA_PSK_EXT.1

No specific audit functions are identified.

### FIA_PSK_EXT.1 Pre-Shared Key Composition

Hierarchical to: No other components.

Dependencies to: FCS_IPSEC_EXT.1 IPsec Protocol

#### FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for IPsec and [**selection**: *IKEv2*, *multifactor authentication filtering*].

#### FIA_PSK_EXT.1.2

The TSF shall be able to accept the following as pre-shared keys: [**selection**: *generated bit-based*, *password-based*, *HMAC-based one-time password*, *time-based one-time password*, *combination of a generated bit-based and HMAC-based one-time password*, *combination of a generated bit-based and time-based one-time password*, *combination of a password-based and HMAC-based one-time password*, *combination of a password-based and time-based one-time password*] keys.

### Management: FIA_PSK_EXT.2

No specific management functions are identified.

### Audit: FIA_PSK_EXT.2

The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- Failure of the randomization process

### FIA_PSK_EXT.2 Generated Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FIA_PSK_EXT.1 Pre-Shared Key Composition

#### FIA_PSK_EXT.2.1

The TSF shall be able to [**selection**:

- *accept externally generated pre-shared keys,*
- *generate [**selection**: 128, 256] bit-based pre-shared keys via FCS_RBG_EXT.1.*

]

### Management: FIA_PSK_EXT.3

No specific management functions are identified.

### Audit: FIA_PSK_EXT.3

The following actions should be auditable if FAU_GEN Security Audit Data Generation is included in the PP/ST:

- Failure of the randomization process

#### FIA_PSK_EXT.3 Password-Based Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FIA_PSK_EXT.1 Pre-Shared Key Composition

##### FIA_PSK_EXT.3.1

The TSF shall support a PSK of up to [**assignment**: *positive integer of 64 or more*] characters.

##### FIA_PSK_EXT.3.2

The TSF shall allow PSKs to be composed of any combination of upper case characters, lower case characters, numbers, and the following special characters: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")", and [**selection**: *[**assignment**: other supported special characters], no other characters*]

##### FIA_PSK_EXT.3.3

The TSF shall perform Password-based Key Derivation Functions in accordance with a specified cryptographic algorithm HMAC- [**selection**: *SHA-256*, *SHA-384*, *SHA-512*], with [**assignment**: *positive integer of 4096 or more*] iterations, and output cryptographic key sizes [**selection**: *128*, *256*] that meet the following: [NIST SP 800-132].

##### FIA_PSK_EXT.3.4

The TSF shall not accept PSKs less than [**selection**: *a value settable by the administrator, [**assignment**: minimum PSK length accepted by the TOE, must be >= 6]*] and greater than the maximum PSK length defined in FIA_PSK_EXT.3.1.

##### FIA_PSK_EXT.3.5

The TSF shall generate all salts using an RBG that meets FCS_RBG_EXT.1 and with entropy of [**assignment**: *value equal to or greater than 128*] bits.

##### FIA_PSK_EXT.3.6

The TSF shall require the PSK to be entered before every initiated connection.

##### FIA_PSK_EXT.3.7

The TSF shall [**selection**: *provide a password strength meter, check the password against a denylist, perform no action to assist the user in choosing a strong password*].

### C.2.4.3 FIA_TOTP_EXT Time-Based One-Time Password Pre-Shared Keys

#### Family Behavior

Components in this family define requirements for the use of Time-Based One-Time password authentication, including generation methods and usage restrictions.

#### Component Leveling

FIA_TOTP_EXT ——— 1

FIA_TOTP_EXT.1, Time-Based One-Time Password Pre-Shared Keys, defines the implementation of TOTP.

#### Management: FIA_TOTP_EXT.1

No specific management functions are identified.

#### Audit: FIA_TOTP_EXT.1

No specific audit functions are identified.

#### FIA_TOTP_EXT.1 Time-Based One-Time Password Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FPT_STM.1 Reliable Time Stamps

##### FIA_TOTP_EXT.1.1

The TSF shall support Time-Based One-Time Password (TOTP) authentication in accordance with RFC 6238 to authenticate the user before establishing VPN connection.

##### FIA_TOTP_EXT.1.2

The TSF shall generate a TOTP seed according to FCS_RBG_EXT.1 of [**selection**: *128*, *256*] bits.

#### FIA_TOTP_EXT.1.3

The TSF shall generate a new TOTP seed for each client.

#### FIA_TOTP_EXT.1.4

The TSF shall use [**selection**: *SHA-1*, *SHA-256*, *SHA-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *160*, *256*, *384*, *512*] to derive a TOTP hash from the TOTP seed and current time provided by NTP.

#### FIA_TOTP_EXT.1.5

The TSF shall truncate the TOTP hash per FIA_TOTP_EXT.1.4 to create a TOTP of [**selection**:

- *administrator configurable character length of at least 6*,
- *preset character length of [**selection, choose one of**: 6, 7, 8, 9, 10]*

].

#### FIA_TOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**: administrator configurable value, [**assignment**: value less than 10]] per minute*,
- *lock the associated account after [**selection**: administrator configurable value, [**assignment**: value less than 10]] failed attempts until [**selection**: an administrator unlocks the account, a configurable time period]*

].

#### FIA_TOTP_EXT.1.7

The TSF shall set a time-step size of [**selection**: *a configurable value*, *[**assignment**: a value less than or equal to 30]*] seconds.

#### FIA_TOTP_EXT.1.8

The TSF shall not validate a drift of more than [**selection**: *a configurable value*, *[**assignment**: a value less than or equal to 3]*] time-steps.

#### FIA_TOTP_EXT.1.9

The TSF shall [**selection**: *allow resynchronization by recording time drift within the limit of FIA_TOTP_EXT.1.8*, *not permit resynchronization*].

## C.2.5 TOE Access (FTA)

This Module defines the following extended components as part of the FTA class originally defined by CC Part 2:

### C.2.5.1 FTA_VCM_EXT VPN Client Management

#### Family Behavior

This family defines requirements for how the TSF interacts with VPN clients in its OE.

#### Component Leveling

FTA_VCM_EXT ——— 1

FTA_VCM_EXT.1, VPN Client Management, requires the TSF to assign private (internal) IP addresses to VPN clients that successfully establish IPsec connections with it.

#### Management: FTA_VCM_EXT.1

No specific management functions are identified.

#### Audit: FTA_VCM_EXT.1

There are no auditable events foreseen.

#### FTA_VCM_EXT.1 VPN Client Management

Hierarchical to: No other components.

Dependencies to: FCS_IPSEC_EXT.1 IPsec Protocol
[FTP_ITC.1 Inter-TSF Trusted Channel, or
FTP_TRP.1 Trusted Path]

**FTA_VCM_EXT.1.1**

The TSF shall assign a private IP address to a VPN client upon successful establishment of a security session.

# Appendix D - Implicitly Satisfied Requirements

This appendix lists requirements that should be considered satisfied by products successfully evaluated against this Module. These requirements are not featured explicitly as SFRs and should not be included in the ST. They are not included as standalone SFRs because it would increase the time, cost, and complexity of evaluation. This approach is permitted by [CC] Part 1, 8.2 Dependencies between components.

This information benefits systems engineering activities which call for inclusion of particular security controls. Evaluation against the PP provides evidence that these controls are present and have been evaluated.

All SFR dependencies in this PP-Module are addressed by appropriate SFRs, either from elsewhere in the PP-Module or inherited from the Base-PP.

# Appendix E - Entropy Documentation and Assessment

The TOE does not require any additional supplementary information to describe its entropy sources beyond the requirements outlined in the Base-PP. As with other Base-PP requirements, the only additional requirement is that the entropy documentation also applies to the specific VPN gateway capabilities of the TOE in addition to the functionality required by the claimed Base-PP.

# Appendix F - Acronyms

| Acronym | Meaning |
|---|---|
| Base-PP | Base Protection Profile |
| CA | Certificate Authority |
| CC | Common Criteria |
| CEM | Common Evaluation Methodology |
| CN | Common Name |
| DH | Diffie-Hellman |
| DN | Distinguished Name |
| EP | Extended Package |
| FP | Functional Package |
| FQDN | Fully Qualified Domain Name |
| ICMP | Internet Control Message Protocol |
| IKE | Internet Key Exchange |
| OE | Operational Environment |
| PBKDF | Password-Based Key Derivation Function |
| PP | Protection Profile |
| PP-Configuration | Protection Profile Configuration |
| PP-Module | Protection Profile Module |
| SA | Security Association |
| SAN | Subject Alternative Name |
| SAR | Security Assurance Requirement |
| SFP | Small Form-Factor Pluggable |
| SFR | Security Functional Requirement |
| ST | Security Target |
| TOE | Target of Evaluation |
| TSF | TOE Security Functionality |
| TSFI | TSF Interface |
| TSS | TOE Summary Specification |
| VPN | Virtual Private Network |
| cPP | Collaborative Protection Profile |

# Appendix G - Bibliography


| Identifier | Title |
|---|---|
| [CC] | Common Criteria for Information Technology Security Evaluation -<br>• Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017.<br>• Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017.<br>• Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017. |
| [ND-SD] | Supporting Document - Mandatory Technical Document - Evaluation Activities for Network Device cPP, Version 2.2, December 2019 |
| [NDcPP] | collaborative Protection Profile for Network Devices, Version 2.2E, March 2020 |