PP-Module for SSL/TLS Inspection Proxies

Version: 1.1
2021-09-10
National Information Assurance Partnership

Revision History

Version	Date	Comment
1.1	2021-09-10	Updates to reflect Github conversion, compatibility with NDcPP v2.2E, and Technical Decisions applied to version 1.0
1.0	2019-08-23	Update release

Contents

1 Introduction

1.1 Overview

The scope of this PP-Module is to describe the security functionality of an SSL/TLS Inspection Proxy (STIP) 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 Netowrk Devices (NDcPP), Version 2.2E

This Base-PP is valid because a STIP is a specific type of network appliance that is able to function as an authorized man-in-the-middle for TLS connections.

This PP-Module is intended to specify the functionality of a network device that includes limited Certification Authority (CA) functionality to issue certificates for the purpose of providing network security services on the underlying plaintext. The device accomplishes this by terminating an intended TLS session between a monitored client and specified external servers. The device instead establishes a TLS session thread consisting of a TLS session between the device and the external server and a second TLS session between the device, acting as the external server, and the client. By replacing the end-to-end TLS session with two TLS sessions terminated at the TOE, the device is able to provide additional security services based on the decrypted plaintext.

A network device meeting this PP-Module may perform additional security services on the plaintext, provide the decrypted payload to external network devices to perform the security services, or do both. These additional security services, whether processed internally or externally, may be performed inline, or passively. If multiple security services are provided, some may be inline, while others are performed passively. This PP-Module does not cover the specific requirements associated with various additional services.

A TOE that conforms to a PP-Configuration containing this PP-Module may be a 'Distributed TOE' as defined in the NDcPP; however, the SSL/TLS Inspection Proxy 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.

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.
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.
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 (PP-Module)	An implementation-independent statement of security needs for a TOE type complementary to one or more Base Protection Profiles.
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.
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.
Target of Evaluation (TOE)	The product under evaluation.

1.2.2 Technical Terms

Attribute	A characterization of an entity (monitored client or the server requested by a monitored client) used in the TLS session establishment policy or the plaintext processing policy implemented by the TOE that describes the entity. Common attributes include IP address, name, and certificates associated to an entity.
Block operation	A high-level operation of the TLS session establishment policy implemented by the TOE that prevents TLS sessions between a monitored client and the server requested by the client.
Bypass operation	A high-level operation of the TLS session establishment policy implemented by the TOE that allows a TLS session between a monitored client and the server requested by the client. Alternatively, an operation of the plaintext processing policy implemented by the TOE to bypass certain inspection processing functional components for plaintext data flows established under the SSL/TLS session establishment policy.
Inspect operation	A high-level operation of the TLS session establishment policy implemented by the TOE that establishes a TLS session thread between a monitored client and a server requested by the monitored client in order to provide security services on the underlying plaintext application data.
Inspection processing functional components	A discrete set of security functions implemented within a single logical component, internal or external to the TOE that provides security services based on a plaintext data flow controlled by the TOE intended to protect a monitored client from defined security threats, or to enforce a defined policy regarding the servers allowed to be accessed by monitored clients.
Monitored Client	A TLS client that uses the TOE as an SSL/TLS Inspection Proxy. This device requires a trust anchor to be installed for the internal CA of the TOE, and makes SSL/TLS requests for services external to the enclave. This client makes SSL/TLS requests to a "requested server" through the TOE.
Requested Server	The target of an SSL/TLS request by a monitored client through the TOE. It is typically a service provider for clients using SSL/TLS. If mutual authentication is to be supported, this device requires a trust anchor to be installed for the internal CA of the TOE.
Secure Sockets Layer/Transport Layer Security (SSL/TLS)	A set of security protocols defined by IETF RFCs to establish a secure point-to-point channel between a client and a server. The secure channel provides confidentiality, integrity and proof of origin to plaintext application data transferred between the client and server. SSL refers to early implementations of the SSL/TLS protocols that are deprecated. TLS refers to current versions of the SSL/TLS protocol.
TLS messages	Specific messages defined by TLS protocol standards. The TLS messages addressed in this PP-Module include TLS handshake messages: Client Hello, Server Hello, Server

	Certificate, Server Key Exchange, Client Key Exchange, Certificate Request, Client Certificate, Client Certificate Verify, Server Finished and Client Finished messages.
TLS session parameters	The parameters of a TLS session established by the TOE for protecting thrutraffic, minimally to include: the negotiated version, negotiated cipher suite, the size of any key exchange values sent or received in key exchange messages, the server certificate received, (a reference to) the server certificate sent, the client certificate received, (a reference to) the client certificate sent, and other negotiated values determined by the TLS handshake that are not fixed for all TLS sessions established.
TLS session thread	A connection negotiated by the TOE consisting of a TLS secure point-to-point channel between a monitored client and the TOE, a TLS secure point-to-point channel between the TOE and the requested server, and any traffic flow containing the underlying application plaintext decrypted from one of the SSL/TLS channels, that is transferred within or between inspection processing functional components controlled by the TOE.

1.3 Compliant Targets of Evaluation

The Target of Evaluation (TOE) may be a single device or a collection of devices that interact with each other to meet the requirements of this PP-Module. Other network devices can be used to supplement inspection of plaintext traffic made available by the TOE. Such external devices will be considered as part of the operational environment, unless they are used to meet the requirements of this PP-Module. Audit, web, or directory servers providing access to certificate validity information generated by the TOE, and intermediate or root certification authorities that issue certificates to the TOE's embedded certification authority are considered part of the operational environment and external to the TOE, but interfaces to these essential services which are required for operation of the TOE will be considered within the TOE boundary. Assurance activities to validate an interface include inspection and exercise of these interfaces using a specific instance of the service (audit server, web server, and external certification authority) implemented within the test environment.

This PP-Module includes some functionality typical of firewalls. In particular, a device meeting this PP-Module is configurable so that it can block or process TLS traffic between monitored clients and requested servers. It's important to note that the device may support TLS connections for remote administration; these TLS connections are distinct from those between the monitored clients and requested servers, and must meet different requirements. In the case of an SSL/TLS inspection proxy, the primary processing is to inspect the TLS traffic. A TOE also has the capability of passing the TLS handshake messages intact to allow end-to-end TLS encrypted traffic between monitored clients and specific servers without providing additional services (bypass the inspection) on decrypted traffic. The decision to drop, process, or bypass traffic is based on IP addresses and ports, as well as on the content of TLS handshake messages, including the certificate of the server, and other characteristics of the traffic that might be available. A device can also determine which additional security services, especially those provided by external network devices, are applied to a particular session based on the plaintext exposed, such as HTTP headers including uniform resource locators (URLs), user passwords, or other sensitive information.

This PP-Module does not require facilitating inspection of mutually authenticated TLS sessions. It does not address the management of clients required to support inspection, nor requirements to avoid monitored clients from discovering the existence of such inspection. Processing to support Certificate Pinning is included as an optional requirement since establishing an inspection point prevents the monitored clients from doing so themselves. Similarly, management of the TOE's certificate trust store is required, since monitored clients cannot block traffic from sites using certificates issued by compromised CA certificates after the traffic is inspected.

1.3.1 TOE Boundary

A STIP is one or more network devices that uses CA functionality to replace an end-to-end TLS session with a TLS session between the STIP and a monitored client and another TLS session between the STIP and the TLS endpoint requested by the monitored client (the requested server). Additional functionality within the same network component as STIP functionality, or via external network devices, can be used to perform network security services, such as performing intrusion detection or providing reputation services on the plaintext traffic made available by the TOE. This functionality, while enabled by the TOE is out of scope. However, protecting and separating traffic flows of plaintext to or between discrete functional components performing such network security services is required and considered within the TOE. If the TOE provides an external interface to plaintext traffic for additional network security services, the entirety of all external processing will be considered a single functional component – the TOE is not responsible for controlling the flow of traffic among external systems.

All functionality described by the SFRs are within the TOE boundary, as is the ability for the TSF to establish secure remote connections with trusted entities in the Operational Environment (OE).

Figure 1: TLS Inspection Infrastructure

As can be seen from this figure, the TOE sits between a monitored client and requested server in order to intercept TLS traffic between them. For connections subject to inspection, the TOE will replace the endto-end TLS session between the monitored client and requested server and establish a TLS session thread in order to forward the plaintext application traffic to one or more inspection processing functional components in the operational environment for inspection. The TSF provides an embedded CA that is used to reconstruct the TLS channel and pass it to its intended destination in an encrypted format. The embedded CA provides certificates it issues to an (external) certificate repository and provides certificate status information to an (internal or external) certificate status presentation mechanism.

1.4 Use Cases

Requirements in this PP-Module are designed to address the security problem in the following use cases. The description of these use cases provide instructions for how the TOE and its OE should be made to support the functionality required by this PP-Module.

This PP-Module permits the inspection of mutually-authenticated TLS sessions between monitored clients and requested servers via exception processing. However, as a best practice, it is recommended instead that this behavior be handled as part of the TLS Inspection Bypass and/or TLS Session Blocking functionality. If the TOE provides inspection processing for mutually authenticated traffic, the ST must claim these optional SFRs.

This PP-Module does not specify routing policies for non-TLS traffic and exception processing should not be used to address functionality otherwise included in the collaborative Protection Profile for Stateful Traffic Filter Firewalls.

[USE CASE 1] Inspection Operation
The TOE intercepts traffic authorized for inspection from monitored clients requesting a serveronly authenticated TLS session with a requested server. The TOE initiates a TLS session with the requested server and validates the requested server's certificate as legitimately issued by a trusted element of its trust store. The TOE authenticates the server on behalf of the client and generates a certificate that indicates the TOE is an authorized proxy for the requested server. The certificate is issued by the TOE's embedded CA, which is trusted by the monitored client. The TOE establishes a valid TLS session with the monitored client using the issued certificate. Any TLS traffic between the monitored client and the requested server is decrypted by the TOE and assigned to a unique TLS session thread that is routed to one or more inspection processes, and encrypted into the respective TLS sessions.

[USE CASE 2] TLS Bypass Operation
The TOE intercepts traffic authorized for inspection from monitored clients requesting a serveronly authenticated TLS session with a requested server. The TOE initiates a TLS session with the requested server and validates the requested server's certificate as legitimately issued by a trusted element of its trust store. The TOE authenticates the server on behalf of the client and generates a certificate that indicates the TOE is an authorized proxy for the requested server. The certificate is issued by the TOE's embedded CA, which is trusted by the monitored client. The TOE establishes a valid TLS session with the monitored client using the issued certificate. Any TLS traffic between the monitored client and the requested server is decrypted by the TOE and assigned to a unique TLS session thread that is routed to one or more inspection processes, and encrypted into the respective TLS sessions.

[USE CASE 3] TLS Blocking Operation
The TOE intercepts traffic authorized for inspection from monitored clients requesting a serveronly authenticated TLS session with a requested server. The TOE initiates a TLS session with the requested server and validates the requested server's certificate as legitimately issued by a trusted element of its trust store. The TOE authenticates the server on behalf of the client and generates a certificate that indicates the TOE is an authorized proxy for the requested server. The certificate is issued by the TOE's embedded CA, which is trusted by the monitored client. The TOE establishes a valid TLS session with the monitored client using the issued certificate. Any TLS traffic between the monitored client and the requested server is decrypted by the TOE and assigned to a unique TLS session thread that is routed to one or more inspection processes, and encrypted into the respective TLS sessions.

[USE CASE 4] Exception Processing

The TOE intercepts traffic authorized for inspection from monitored clients requesting a serveronly authenticated TLS session with a requested server. The TOE initiates a TLS session with the requested server and validates the requested server's certificate as legitimately issued by a trusted element of its trust store. The TOE authenticates the server on behalf of the client and generates a certificate that indicates the TOE is an authorized proxy for the requested server. The certificate is issued by the TOE's embedded CA, which is trusted by the monitored client. The TOE establishes a valid TLS session with the monitored client using the issued certificate. Any TLS traffic between the monitored client and the requested server is decrypted by the TOE and assigned to a unique TLS session thread that is routed to one or more inspection processes, and encrypted into the respective TLS sessions.

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).

> No PPs or PP-Modules are allowed to be specified in a PP-Configuration with this PP-Module aside from its supported Base-PP.

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 Protection Profile.

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, and any organizational security policies that the TOE is expected to enforce.

A STIP is a network device that embeds limited CA functionality to support the replacement of end-to-end TLS sessions with TLS session threads, making the underlying plaintext available to additional network security functionality. As such, it exposes data within the TOE boundary, and to external processes, which would normally be encrypted. It manages a CA signing key that is trusted by the monitored clients to issue TLS server certificates representing the requested servers for which inspection is authorized.

The proper installation and configuration of the VPN client is critical to its correct operation such that proper handling of the TOE by an administrator is also addressed.

Note that as a PP-Module, all threats, assumptions, and OSPs defined in the Base-PP will also apply to a TOE unless otherwise specified, depending on which of the Base-PPs it extends. The Security Functional Requirements defined in this PP-Module will mitigate the threats that are defined in the PP-Module but may also mitigate some threats defined in the Base-PPs in more comprehensive detail due to the specific capabilities provided by a VPN client.

3.1 Threats

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

T.UNTRUSTED_COMMUNICATION
Untrusted intermediate systems have access to provide unauthorized communications to the TOE, or to manipulate authorized TLS messages in an attempt to compromise the TOE, the monitored clients, or the requested servers. Within this PP-Module, the focus is on an adversary that controls or exploits a requested server that may attempt to cause the device to inappropriately bypass inspection.

Use of weak cryptography can allow adversary access to plaintext intended by the monitored clients to be encrypted. Such access could disclose user passwords that facilitate additional activities against users of monitored clients. Within this PP-Module, the focus is on the use of weak cryptography and adversary attempts to degrade the cryptographic operations within the TLS protocol.

External network security devices may communicate with the TOE to apply security services to the exposed plaintext. An adversary may attempt to gain access the plaintext via misrouting of traffic or manipulate the traffic in such a way as to cause unauthorized exposure, denial of service, or corruption of the underlying plaintext.

T.AUDIT
Certificates issued by the device are trusted by monitored clients, and are required for analysis if traffic processed by the device causes the client to fail or become compromised. Unknown activity related to the issuance and use of certificates can allow an adversary to mask client exploits through or via the TOE, especially if the device fails before the incident can be understood. Unknown activity associated to routing configurations, communications with the TOE, as well as the decision to bypass inspection of traffic can allow an adversary to mask attempts to access monitored clients.

T.UNAUTHORIZED_USERS
In addition to managing administrative credentials, authorized users may have role restrictions to limit their access to the device's certification authority functionality. In addition to the threat of disclosure or modification of authorized user credentials to users without authorized access to the device, a user with limited access might attempt to extend their access by gaining access to other user's credentials.

T.CREDENTIALS
In addition to device credentials used in protected communications, the device maintains a trusted certification authority signing key. Any disclosure or unauthorized manipulation of the signing key can result in unintended certificates, signed executable, or signed data that would be trusted by monitored clients. Any modification of the signing key can result in denial of service to inspection capabilities, or to the monitored clients.

T.SERVICES
Manipulation of the device can result in issued certificates being used for unauthorized purposes or abuse of inspection services. An authorized user (AU) (or adversary able to gain access to AU credentials) can access or misuse device services, or disclose sensitive or security critical data.

T.DEVICE_FAILURE
Failure of the certification authority component can result in unauthorized or improperly constrained certificates, or the inability to properly manage the validity of issued certificates. Failure of routing traffic to inspection processing (internal or external) can result in unauthorized disclosure or modification of traffic, or denial of service to monitored clients.

T.UNAUTHORIZED_DISCLOSURE
In addition to general threats to network devices, the TOE controls access to sensitive data that is intended by the monitored client to be encrypted.

T.INAPPROPRIATE_ACCESS
Decryption services applied to traffic between monitored clients and unintended servers can violate privacy laws, or disclose unauthorized traffic to inspection processes. Certification authority signature applied to unauthorized data could facilitate adversary exploits of monitored clients.

3.2 Assumptions

These assumptions are made on the Operational Environment 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 Operational Environment that does not meet these assumptions, the TOE may no longer be able to provide all of its security functionality.

All assumptions for the operational environment of the Base-PP also apply to this PP-Module.

A.LIMITED_FUNCTIONALITY is still operative, but the assumed functionality of the TOE includes the behavior needed to satisfy the functional claims of 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.TRUSTED_ADMINISTRATOR is still operative, but the functional claims of this PP-Module offer a limited ability to protect against malicious administrators, which is not within the scope of the original assumption.

A.RESIDUAL_INFORMATION is still operative, but the residual information is expanded to include information relevant to STIP operation (e.g. decrypted SSL/TLS payload, ephemeral keys). This document does not define any additional assumptions.

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.

P.AUTHORIZATION_TO_INSPECT
The authority to inspect client traffic may be limited by law, regulation, or policies based on the monitored client, requested server, or nature of the traffic.

4 Security Objectives

4.1 Security Objectives for the TOE

O.AUDIT_LOSS_RESPONSE

The TOE will respond to possible loss of audit records when an audit trail cannot be written to by restricting auditable events.

Addressed by: FAU_STG.4

O.AUDIT_PROTECTION

The TOE will protect audit records against unauthorized access, modification, or deletion to ensure accountability of user actions.

Addressed by: FAU_STG.1 (from Base-PP), FAU_SAR.1 (optional)

O.CERTIFICATES

The TSF must ensure that certificates, certificate revocation lists, and certificate status information are valid.

Addressed by: FIA_X509_EXT.1/Rev (from Base-PP), FIA_X509_EXT.3 (from Base-PP), FDP_CER_EXT.1, FDP_CER_EXT.2, FDP_CER_EXT.3, FDP_CSIR_EXT.1, FIA_ENR_EXT.1, FIA_X509_EXT.1/STIP, FIA_X509_EXT.2/STIP, FDP_PIN_EXT.1 (optional), FIA_ESTC_EXT.2 (optional), FDP_CER_EXT.4 (selection-based), FDP_CER_EXT.5 (selection-based), FDP_CRL_EXT.1 (selection-based), FDP_CSI_EXT.1 (selection-based), FDP_CSI_EXT.2 (selection-based), FDP_OCSP_EXT.1 (selection-based), FDP_OCSPS_EXT.1 (selection-based), FIA_ESTC_EXT.1 (selection-based)

O.DISPLAY_BANNER

The TOE will display an advisory warning regarding use of the TOE.

Addressed by: FTA_TAB.1 (from Base-PP), FTA_TAB.1/TLS (selection-based)

O.PERSISTENT_KEY_PROTECTION

The TOE will provide appropriate confidentiality and access protection to persistent keys and security critical parameters stored by the TOE.

Addressed by: FCS_STG_EXT.1, FDP_STG_EXT.1, FPT_KST_EXT.1, FPT_KST_EXT.2, FCS_CKM_EXT.5 (selection-based)

O.PROTECTED_COMMUNICATIONS

The TOE will provide protected communication channels for administrators, other parts of a distributed TOE, and authorized IT entities. The TOE will protect data assets when they are being transmitted to and from the TOE, including through intervening untrusted components.

Addressed by: FCS_CKM.4 (from Base-PP), FCS_TLSC_EXT.1 (from Base-PP), FCS_TLSS_EXT.1 (from Base-PP), FTP_ITC.1 (refined from Base-PP), FCS_COP.1/STIP, FCS_TTTC_EXT.1, FCS_TTTC_EXT.5, FCS_TTTS_EXT.1, FDP_PPP_EXT.1, FDP_PRC_EXT.1, FDP_STIP_EXT.1, FDP_TEP_EXT.1, FCS_TTTC_EXT.3 (selection-based), FCS_TTTC_EXT.4 (selection-based), FCS_TTTS_EXT.3 (selection-based), FCS_TTTS_EXT.4 (selection-based), FDP_STIP_EXT.2 (selection-based)

O.RECOVERY

The TOE will have the ability to store and recover to a previous state at the direction of the administrator (e.g., provide support for archival and recovery capabilities).

Addressed by: FPT_FLS.1, FPT_RCV.1

O.RESIDUAL_INFORMATION_CLEARING

The TOE will ensure that any data contained in a protected resource is not available when the resource is reallocated.

Addressed by: FDP_RIP.1

O.SYSTEM_MONITORING

The TOE will provide the ability to generate audit data and send that data to an external IT entity. The TOE will record in audit records: date and time of action and the entity responsible for the action. The TOE will provide the ability to store and review certificate information.

Addressed by: FAU_STG_EXT.1 (from Base-PP), FAU_GEN.1/STIP, FAU_GCR_EXT.1, FAU_SAR.3 (optional), FAU_SCR_EXT.1 (selection-based)

O.TOE_ADMINISTRATION

The TOE will provide mechanisms to ensure that only privileged users are able to log in and configure the TOE, and provide protections for logged-in users. The TOE will ensure that administrative responsibilities are separated across different roles in order to mitigate the impact of improper

administrative activities or unauthorized administrative access. Addressed by: FMT_MOF.1, FMT_SMF.1/STIP, FMT_SMR.2/STIP

4.2 Security Objectives for the Operational Environment

The Operational Environment 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 Operational Environment consist of a set of statements describing the goals that the Operational Environment 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. The Operational Environment 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). This section defines the security objectives that are to be addressed by the IT domain or by nontechnical or procedural means. As indicated above, if requirements supporting an objective on the TOE (in the previous table) are implemented in whole or in part by the platform, the ST should indicate this by an entry in this table with that objective.

All security objectives for the operational environment 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.RESIDUAL_INFORMATION is still operative, but the residual information is expanded to include information relevant to STIP operation (e.g. decrypted SSL/TLS payload, ephemeral keys).

OE.AUDIT
>The operational environment includes an audit server with adequate storage to retain the audit record, and the audit server provides adequate availability, integrity, and access control to the audit record to support operational requirements. Administration of the audit server is separate from that of the SSL/TLS inspection proxy, and can support all required role separations.
>
>*Rationale: Each certificate issued by the SSL/TLS inspection device is trusted by monitored clients for the validity period asserted in the certificate. Inappropriate issuance or use of certificates issued by the SSL/TLS inspection device may require the circumstances of the issuance to be investigated, and appropriate actions (e.g., certificate revocation, administrative actions, etc.) to be taken.*

OE.CERT_REPOSITORY
>The OE provides a certificate repository for storage of certificates (and optionally CRLs) issued by the TSF.
>
>*Rationale: Each certificate issued by the SSL/TLS inspection device is trusted by monitored clients for the validity period asserted in the certificate. Inappropriate issuance of certificates, especially certificates with code-signing or which can act as subordinate CAs to issue additional certificates, or inappropriate use of certificates issued by the SSL/TLS inspection device to conduct unauthorized inspection, or to gain access to protected resources may require the circumstances of the issuance to be investigated, and appropriate actions (e.g., certificate revocation, administrative actions, etc.) to be taken.*

OE.CERT_REPOSITORY_SEARCH
>The OE provides the ability to search a certificate repository for specific certificate fields in certificates issued by the TSF and return the certificate and an identifier for the certificate that can be used to search the audit trail for events related to that certificate and for unauthorized or improperly constrained certificates.
>
>*Rationale: Each certificate issued by the SSL/TLS inspection device is trusted by monitored clients for the validity period asserted in the certificate. Inappropriate issuance or use of certificates issued by the SSL/TLS inspection device may require the circumstances of the issuance to be investigated, and appropriate actions (e.g., certificate revocation, administrative actions, etc.) to be taken.*

4.3 Security Objectives Rationale

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

<div align="center">

Table 1: Security Objectives Rationale

</div>

Threat, Assumption, or OSP	Security Objectives	Rationale
T.UNTRUSTED_COMMUNICATION	O.PROTECTED_COMMUNICATIONS	Data traversing the TOE is subject to authenticity and integrity verification.
T.AUDIT	O.AUDIT_LOSS_RESPONSE	The TOE provides mechanisms to deal with audit trails being unavailable.
	O.AUDIT_PROTECTION	Audit records are protected from modification, deletion, and unauthorized access.
	O.SYSTEM_MONITORING	Audit records contain the

		information necessary to determine cause for concerns.
	OE.AUDIT	Storage within an external audit server provides increased record capacity.
	OE.CERT_REPOSITORY	The certificate repository provides a comprehensive set of certificates generated by the TOE that can be searched.
	OE.CERT_REPOSITORY_SEARCH	Ability to search the audit trail for certificate related events provides confidence in certificate validity and proper use.
T.UNAUTHORIZED_USERS	O.TOE_ADMINISTRATION	Use of role separation and authentication mechanisms ensure that only authorized users can access the TOE.
T.CREDENTIALS	O.CERTIFICATES	The TOE tracks certificates, certificate revocation lists, and certificate status information used by the TSF.
	O.PERSISTENT_KEY_PROTECTION	Keys stored on the TOE are protected from unauthorized use and disclosure.
	OE.CERT_REPOSITORY	A certificate repository for all certificates issued by the TOE is provided, making verification straightforward.
T.SERVICES	O.CERTIFICATES	The TOE verifies certificates, certificate revocation lists, and certificate status information prior to any use.
	O.PROTECTED_COMMUNICATIONS	Data traversing the TOE is subject to authenticity, confidentiality, and integrity verification.
	O.TOE_ADMINISTRATION	Use of role separation and authentication mechanisms mitigates the risk of misuse and improper disclosure.
T.DEVICE_FAILURE	O.CERTIFICATES	The TOE verifies certificates, certificate revocation lists, and certificate status information is valid.
	O.INTEGRITY_PROTECTION	Software, TSF, and user data are protected via integrity mechanisms.
	O.PERSISTENT_KEY_PROTECTION	Keys stored on the TOE are protected from unauthorized use and

		disclosure.
	O.RECOVERY	Administrators have the ability to restore the TOE to a previous (known-good) state.
T.UNAUTHORIZED_DISCLOSURE	O.PROTECTED_COMMUNICATIONS	Data traversing the TOE is subject to authenticity, confidentiality, and integrity verification.
	O.TOE_ADMINISTRATION	Use of role separation and authentication mechanisms mitigates the risk of misuse and ensures the device is properly managed.
T.INAPPROPRIATE_ACCESS	O.RESIDUAL_INFORMATION_CLEARING	The TOE's lack of residual data retention ensures that unauthorized access to information is not possible.
	O.TOE_ADMINISTRATION	Use of role separation and authentication mechanisms mitigates the risk of misuse and improper disclosure.
	OE.RESIDUAL_INFORMATION	Sensitive information residing within the operational environment, such as keys and decrypted data, are unavailable.
P.AUTHORIZATION_TO_INSPECT	O.DISPLAY_BANNER	The TOEs advisory warning includes consent to monitor.
	O.PROTECTED_COMMUNICATIONS	The TSF ensures that data traversing the TOE boundary is protected, alleviating concerns about inspection.
	O.TOE_ADMINISTARTION	Administrator roles provide separation of activities and ensure inspection is authorized and performed properly.

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 General Purpose Operating Systems PP Security Functional Requirements Direction

In a PP-Configuration that includes the NDcPP, the STIP 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 General Purpose Operating Systems PP and relevant to the secure operation of the TOE.

5.1.1.1 Security Audit (FAU)

FAU_STG.1 Protected Audit Trail Storage

FAU_STG.1.1

This PP-Module does not modify this SFR as it is defined in the Base-PP. This SFR is optional in the Base-PP but is mandatory for a TOE that conforms to this PP-Module.

Evaluation Activities ▼

> *FAU_STG.1*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

5.1.1.2 Cryptographic Support (FCS)

FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1

The TSF shall destroy all cryptographic keys **and critical security parameters, when no longer required** in accordance with the specified cryptographic key destruction method [**selection**:

- *For plaintext keys in volatile storage, the destruction shall be executed by a [**selection**:*
    - *Single overwrite consisting of [**selection**:*
        - *a pseudo-random pattern using the TSF's RBG,*
        - *zeroes,*
        - *ones,*
        - *a new value of the key,*
        - *[**assignment**: a static or dynamic value that does not contain any CSP]*

        ],
    - *Destruction of reference to the key directly followed by a request for garbage collection*

    ],
- *For plaintext keys in non-volatile storage, the destruction shall be executed the invocation of an interface provided by a part of the TSF that [**selection**:*
    - *Logically addresses the storage location of the key and performs a [**selection**: single, [**assignment**: number of passes]-pass] overwrite*

consisting of [**selection**:
- *a pseudo-random pattern using the TSF's RBG,*
- *zeroes,*
- *ones,*
- *a new value of the key,*
- [**assignment**: *a static or dynamic value that does not contain any CSP*]

],
- *Instructs a part of the TSF to destroy the abstraction that represents the key*

]

] that meets the following: [*no standard*].

Application Note: This SFR is refined from its definition in the Base-PP through the inclusion of security critical parameters and clarifies when destruction is required; a STIP device includes persistent keys, including the embedded CA's signing private key that should not be destroyed until they are no longer needed. Security critical parameters includes security related information (e.g., secret and private cryptographic keys, authentication data such as passwords and PINs) appearing in plaintext or otherwise unprotected form and whose disclosure or modification can compromise the security of a CA or the security of the information protected by the CA or the security of the information protected by the CA.

Evaluation Activities ▽

> *FCS_CKM.4*
> *This SFR is refined in this PP-Module to include requirementsfor destruction of security critical parameters as well as keys. The EA for the Base-PP are extended to include security critical parameters whenever keys are indicated.*

FCS_TLSC_EXT.1 TLS Client Protocol Without Mutual Authentication

FCS_TLSC_EXT.1.1

This PP-Module does not modify this SFR as it is defined in the Base-PP. This SFR is selection-based in the Base-PP but is mandatory for a TOE that conforms to this PP-Module because of this PP-Module's modifications to FTP_ITC.1.

Evaluation Activities ▽

> *FCS_TLSC_EXT.1*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication

FCS_TLSS_EXT.1.1

This PP-Module does not modify this SFR as it is defined in the Base-PP. This SFR is selection-based in the Base-PP but is mandatory for a TOE that conforms to this PP-Module because of this PP-Module's modifications to FTP_ITC.1.

Evaluation Activities ▽

> *FCS_TLSS_EXT.1*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

5.1.1.3 Identification and Authentication

FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev

This PP-Module does not modify this SFR as it is defined in the Base-PP. This SFR is selection-based in the Base-PP but is mandatory for a TOE that conforms to this PP-Module because of this PP-Module's modifications to FTP_ITC.1.

Application Note: FIA_X509_EXT.1/STIP defines the TOE's X.509 validation behavior for TLS certificates presented to the TSF as part of TLS proxying. At minimum, FIA_X509_EXT.1/Rev is used by the TOE to validate any certificates

loaded onto it. If the TOE has other functions that require the use of X.509 certificates (e.g. code signing for integrity testing or software updates, TLS interfaces used for a purpose other than session proxying such as audit server or authentication server connections), FIA_X509_EXT.1/Rev applies to those as well.

Evaluation Activities ▼

> *FIA_X509_EXT.1/Rev*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

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

FIA_X509_EXT.2.2

When the TSF cannot establish a connection to determine the revocation status of a certificate, the TSF shall [**selection**:

- *allow the [selection: Security Administrator, CA Operations Staff] to choose whether to [selection: accept the certificate, associate the failed connection event per FDP_TEP_EXT.1.5] in these cases,*
- *accept the certificate,*
- *not accept the certificate*

].

Application Note: "TLS" is moved outside the selection in the first element, since the TOE must implement TLS to accomplish the STIP functionality. The application notes for the first element from the Base-PP also apply.

It is worth noting that since this SFR applies to all uses of certificates in the TOE, it may be the case that the actions taken in response to a failure to be able to determine revocation status (which is specified in the 2nd element) is handled differently for different connections. If this is the case, the ST author must make it clear which actions are associated with which connections so that the correct evaluation of the functionality can be performed.

The second element has three modifications from that in the Base-PP. First, the word "validity" is replaced with "revocation status" for clarity. This is consistent with what is in the application note in the NDcPP, and using "revocation status" more directly indicates what is required.

Second, the general notion of "administrator" is replaced with the more refined roles defined in this PP-Module; the ST author should make the appropriate selection.

Finally, a selection is added that allows ST author flexibility in addressing the issue of failure to connect to check revocation status in the specific case that the certificates being checked are associated with either a monitored client or a requested server. This selection ("to associate the failed connection event per FDP_TEP_EXT.1.5"), when chosen, indicates that selected administrative role is able to specify a STIP operation (block, bypass, inspect) to be taken in the event that the revocation status can't be checked. The requirement that the TOE be able to perform this operation when such an event occurs is specified in FDP_TEP_EXT.1.5.

Evaluation Activities ▼

> *FIA_X509_EXT.2*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1

In the Base-PP, this SFR is optional but must be claimed in any situation where the TOE presents its own X.509 certificate to an external entity (e.g. any case where the TOE acts as a TLS server or where the TOE acts as a TLS client in an

connection that uses mutual authentication). A STIP TOE must present an X.509 certificate to an external entity as part of TLS session proxying. The TOE may obtain this certificate either using PKCS#10 (covered by this SFR) or through Enrollment over Secure Transport (EST), which is covered by the selection-based SFR FIA_ESTC_EXT.1. Therefore, the ST author only claims FIA_X509_EXT.3 if PKCS#10 is selected in FIA_ENR_EXT.1.

Evaluation Activities ▼

> *FIA_X509_EXT.3*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

5.1.1.4 Trusted Path/Channels

FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1

> The TSF shall be capable of using **TLS** and [**selection**: *IPsec*, *SSH*, *DTLS*, *HTTPS*, ***no other protocols***] to provide a trusted communication channel between itself and authorized IT entities supporting the following capabilities: audit server, **TLS session proxying**, [**selection**: *authentication server*, [***assignment**: other capabilities*], *no other capabilities*] that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from disclosure and detection of modification of the channel data.

FTP_ITC.1.2

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

FTP_ITC.1.3

> The TSF shall initiate communication via the trusted channel for **establishment of TLS proxy connections,** [**assignment**: *list of services for which the TSF is able to initiate communications*].

Evaluation Activities ▼

> *FTP_ITC.1*
> *There is no change to the Base-PP EAs for this SFR when this PP-Module is claimed.*

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
FCS_CKM.1/VPN	No events specified	
FCS_IPSEC_EXT.1	Decisions to DISCARD or BYPASS network packets processed by the TOE.	Presumed identity of source subject. The entry in the SPD that applied to the decision.
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Identity of destination subject. Reason for failure.
FCS_IPSEC_EXT.1	Establishment/Termination of an IPsec SA.	Identity of destination subject. Transport layer protocol, if applicable. Source subject service identifier, if applicable. Non-TOE endpoint of connection (IP address) for both successes and failures.
FDP_RIP.2	No events specified	
FMT_SMF.1/VPN	Success or failure of management	

| FPT_TST_EXT.1/VPN | No events specified | |

5.2.2 Cryptographic Support (FCS)

FCS_CKM.1/VPN VPN Cryptographic Key Generation (IKE)

FCS_CKM.1.1/VPN

The TSF shall **[selection: *invoke platform-provided functionality, implement functionality*] to** generate **asymmetric** cryptographic keys **used for IKE peer authentication** in accordance with: **[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-256, P-384 and [selection: P-521, no other curves]***

] and specified cryptographic key sizes [*equivalent to, or greater than, a symmetric key strength of 112 bits*] ~~that meet the following:~~ [**assignment**: *list of standards*].

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 entities during the IKE (either v1 or v2) key exchange. 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 Certificate Authority in the Operational Environment.

As indicated in FCS_IPSEC_EXT.1, the TOE is required to implement support for RSA or ECDSA (or both) for authentication.

See NIST Special Publication 800-57, "Recommendation for Key Management" for information about equivalent key strengths.

Evaluation Activities ▽

FCS_CKM.1/VPN
TSS
The evaluator shall examine the TSS to verify that it describes how the key generation functionality is invoked.
Guidance
There are no AGD EAs for this requirement.
Tests
If this functionality is implemented by the TSF, refer to the following EAs, depending on the TOE's claimed Base-PP:

- *GPOS PP: FCS_CKM.1*
- *MDF PP: FCS_CKM.1*
- *App PP: FCS_CKM.1(1)*
- *MDM PP: FCS_CKM.1*

FCS_IPSEC_EXT.1 IPsec

FCS_IPSEC_EXT.1.1

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

Application Note: In the following elements of the FCS_IPSEC_EXT.1 component, it is allowable for some or all of the individual elements to be implemented by the platform on which the VPN client operates. However, this is only the case when the platform is within the TOE boundary, as is the case where this PP-Module is being claimed on top of a general-purpose operating system or a mobile device.

When the TOE is a standalone software application, the IPsec functionality must be implemented by the TSF, though it is permissible for the TSF to invoke cryptographic algorithm services from the TOE platform to support the TOE's implementation of IPsec. The TOE may also rely on the TOE platform for X.509 certificate validation services, though it is the responsibility of the TSF to take the proper action based on the validation response that is returned.

It is also permissible for the TSF to rely on low-level capabilities of the platform to perform enforcement and routing functions as a result of the policies the TSF

maintains. For example, while the TSF must provide the capability to implement the Security Policy Database abstraction, it is allowed for the TSF to depend on the platform-provided network stack/driver to perform the low-level packet filtering and routing actions once the TSF has set up those rules as defined by the SPD.

While enforcement of the IPsec requirements must be implemented by the TSF, it is permissible for the TSF to receive configuration of the IPsec behavior from an environmental source, most notably a VPN gateway.

RFC 4301 calls for an IPsec implementation to protect IP traffic through the use of a Security Policy Database (SPD). The SPD is used to define how IP packets are to be handled: PROTECT the packet (e.g., encrypt the packet), BYPASS the IPsec services (e.g., no encryption), or DISCARD the packet (e.g., drop the packet). The SPD can be implemented in various ways, including router access control lists, firewall rulesets, a "traditional" SPD, etc. Regardless of the implementation details, there is a notion of a "rule" that a packet is "matched" against and a resulting action that takes place.

While there must be a means to order the rules, a general approach to ordering is not mandated, as long as the TOE can distinguish the IP packets and apply the rules accordingly. There may be multiple SPDs (one for each network interface), but this is not required.

A VPN gateway fully implements the IPsec capability and provides an administrative interface to establish and populate an SPD. A VPN client is not required to provide an administrative interface to create or maintain an SPD.

As an alternative, a client may provide an interface that can be used by another application or network entity, such as a VPN gateway, as a means to establish and populate the SPD. In either of these cases (the client provides an administrative interface, or an API), while the client is expected to maintain the SPD abstraction, it is permitted for the low-level enforcement and routing activities to be implemented by platform capabilities (e.g., a network driver) as configured by the client.

FCS_IPSEC_EXT.1.2

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

Application Note: If the TOE is used to connect to a VPN gateway for the purposes of establishing a secure connection to a private network, the ST author is expected to select tunnel mode. If the TOE uses IPsec to establish an end-to-end connection to another IPsec VPN Client, the ST author is expected to select transport mode. If the TOE uses IPsec to establish a connection to a specific endpoint device for the purpose of secure remote administration, the ST author is expected to select transport mode.

FCS_IPSEC_EXT.1.3

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

FCS_IPSEC_EXT.1.4

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

Application Note: If this functionality is configurable, the TSF may be configured by a VPN Gateway or by an Administrator of the TOE itself.

FCS_IPSEC_EXT.1.5

The The TSF shall implement the protocol: [**selection**:
- *IKEv1, using Main Mode for Phase I exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109,* [**selection**: *no other RFCs for extended sequence numbers, RFC 4304 for extended sequence numbers*], [**selection**: *no other RFCs for hash functions, RFC 4868 for hash functions*], and [**selection**: *support for XAUTH, no support for XAUTH*],
- *IKEv2 as defined in RFC 7296 (with mandatory support for NAT traversal as specified in section 2.23), RFC 8784, RFC 8247, and* [**selection**: *no other RFCs for hash functions, RFC 4868 for hash functions*]
].

FCS_IPSEC_EXT.1.6

The TSF shall ensure the encrypted payload in the [**selection**: *IKEv1, IKEv2*]

protocol uses the cryptographic algorithms [*AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and* [**selection**: *AES-GCM-128 as specified in RFC 5282*, *AES-GCM-256 as specified in RFC 5282*, *no other algorithm*].

Application Note: If this functionality is configurable, the TSF may be configured by a VPN Gateway or by an Administrator of the TOE itself.

FCS_IPSEC_EXT.1.7

The TSF shall ensure that [**selection**:
- *IKEv2 SA lifetimes can be configured by [**selection**: an Administrator, a VPN Gateway] based on [**selection**: number of packets/number of bytes, length of time]* ,
- *IKEv1 SA lifetimes can be configured by [**selection**: an Administrator, a VPN Gateway] based on [**selection**: number of packets/number of bytes, length of time]* ,
- *IKEv1 SA lifetimes are fixed based on [**selection**: number of packets/number of bytes, length of time]*

]. If length of time is used, it must include at least one option that is 24 hours or less for Phase 1 SAs and 8 hours or less for Phase 2 SAs.

Application Note: The ST author is afforded a selection based on the version of IKE in their implementation. There is a further selection within this selection that allows the ST author to specify which entity is responsible for "configuring" the life of the SA. An implementation that allows an administrator to configure the client or a VPN gateway that pushes the SA lifetime down to the client are both acceptable.

As far as SA lifetimes are concerned, the TOE can limit the lifetime based on the number of bytes transmitted, or the number of packets transmitted. Either packet-based or volume-based SA lifetimes are acceptable; the ST author makes the appropriate selection to indicate which type of lifetime limits are supported.

The ST author chooses either the IKEv1 requirements or IKEv2 requirements (or both, depending on the selection in FCS_IPSEC_EXT.1.5. The IKEv1 requirement can be accomplished either by providing Authorized Administrator-configurable lifetimes (with appropriate instructions in documents mandated by AGD_OPE), or by "hard coding" the limits in the implementation. For IKEv2, there are no hardcoded limits, but in this case it is required that an administrator be able to configure the values. In general, instructions for setting the parameters of the implementation, including lifetime of the SAs, should be included in the operational guidance generated for AGD_OPE. It is appropriate to refine the requirement in terms of number of MB/KB instead of number of packets, as long as the TOE is capable of setting a limit on the amount of traffic that is protected by the same key (the total volume of all IPsec traffic protected by that key).

FCS_IPSEC_EXT.1.8

The TSF shall ensure that all IKE protocols implement DH Groups [*19 (256-bit Random ECP), 20 (384-bit Random ECP), and* [**selection**: *24 (2048-bit MODP with 256-bit POS)*, *15 (3072-bit MODP)*, *14 (2048-bit MODP)*, *no other DH groups*]].

Application Note: The selection is used to specify additional DH groups supported. This applies to IKEv1 and IKEv2 exchanges. It should be noted that if any additional DH groups are specified, they must comply with the requirements (in terms of the ephemeral keys that are established) listed in FCS_CKM.1.

Since the implementation may allow different Diffie-Hellman groups to be negotiated for use in forming the SAs, the assignments in FCS_IPSEC_EXT.1.9 and FCS_IPSEC_EXT.1.10 may contain multiple values. For each DH group supported, the ST author consults Table 2 in 800-57 to determine the "bits of security" associated with the DH group. Each unique value is then used to fill in the assignment (for 1.9 they are doubled; for 1.10 they are inserted directly into the assignment). For example, suppose the implementation supports DH group 14 (2048-bit MODP) and group 20 (ECDH using NIST curve P-384). From Table 2, the bits of security value for group 14 is 112, and for group 20 it is 192. For FCS_IPSEC_EXT.1.9, then, the assignment would read "[224, 384]" and for FCS_IPSEC_EXT.1.10 it would read "[112, 192]" (although in this case the requirement should probably be refined so that it makes sense mathematically).

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) number(s) of bits that is at least twice the "bits of security" value associated with the negotiated Diffie-Hellman group as listed in Table 2 of NIST SP 800-57*,

Recommendation for Key Management – Part 1: General] bits.

FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in IKE exchanges in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^[**assignment**: *(one or more) "bits of security" value(s) associated with the negotiated Diffie-Hellman group as listed in Table 2 of NIST SP 800-57, Recommendation for Key Management – Part 1: General*].

FCS_IPSEC_EXT.1.11

The TSF shall ensure that all IKE protocols perform peer authentication using a [**selection**: *RSA*, *ECDSA*] that use X.509v3 certificates that conform to RFC 4945 and [**selection**: *Pre-shared keys*, *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 (for example, 2409 specifies three authentication methods using public keys; each one supported will be described in the TSS).

If "pre-shared keys" is selected, the selection-based requirement FIA_PSK_EXT.1 must be claimed.

FCS_IPSEC_EXT.1.12

The TSF shall not establish an SA if the [[**selection**: *IP address*, *Fully Qualified Domain Name (FQDN)*, *user FQDN*, *Distinguished Name (DN)*] and [**selection**: *no other reference identifier type*, *[**assignment**: other supported reference identifier types]*]] contained in a certificate does not match the expected value(s) for the entity attempting to establish a connection.

Application Note: The TOE must support at least one of the following identifier types: IP address, Fully Qualified Domain Name (FQDN), user FQDN, or Distinguished Name (DN). In the future, the TOE will be required to support all of these identifier types. The TOE is expected to support as many IP address formats (IPv4 and IPv6) as IP versions supported by the TOE in general. The ST author may assign additional supported identifier types in the second selection.

FCS_IPSEC_EXT.1.13

The TSF shall not establish an SA if the presented identifier does not match the configured reference identifier of the peer.

Application Note: At this time, only the comparison between the presented identifier in the peer's certificate and the peer's reference identifier is mandated by the testing below. However, in the future, this requirement will address two aspects of the peer certificate validation: 1) comparison of the peer's ID payload to the peer's certificate which are both presented identifiers, as required by RFC 4945 and 2) verification that the peer identified by the ID payload and the certificate is the peer expected by the TOE (per the reference identifier). At that time, the TOE will be required to demonstrate both aspects (i.e. that the TOE enforces that the peer's ID payload matches the peer's certificate which both match configured peer reference identifiers).

Excluding the DN identifier type (which is necessarily the Subject DN in the peer certificate), the TOE may support the identifier in either the Common Name or Subject Alternative Name (SAN) or both. If both are supported, the preferred logic is to compare the reference identifier to a presented SAN, and only if the peer's certificate does not contain a SAN, to fall back to a comparison against the Common Name. In the future, the TOE will be required to compare the reference identifier to the presented identifier in the SAN only, ignoring the Common Name.

The configuration of the peer reference identifier is addressed by FMT_SMF.1.1/VPN.

FCS_IPSEC_EXT.1.14

The [**selection**: *TSF*, *VPN Gateway*] 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: If this functionality is configurable, the TSF may be configured by a VPN Gateway or by an Administrator of the TOE itself

The ST author chooses either or both of the IKE selections based on what is implemented by the TOE. Obviously, the IKE version(s) chosen should be consistent not only in this element, but with other choices for other elements in this component. While it is acceptable for this capability to be configurable, the default configuration in the evaluated configuration (either "out of the box" or by configuration guidance in the AGD documentation) must enable this functionality.

Evaluation Activities ▼

FCS_IPSEC_EXT.1
TSS
In addition to the TSS EAs for the individual FCS_IPSEC_EXT.1 elements below, the evaluator shall perform the following:

If the TOE boundary includes a general-purpose operating system or mobile device, the evaluator shall examine the TSS to ensure that it describes whether the VPN client capability is architecturally integrated with the platform itself or whether it is a separate executable that is bundled with the platform.

Guidance
In addition to the Operational Guidance EAs for the individual FCS_IPSEC_EXT.1 elements below, the evaluator shall perform the following:

If the configuration of the IPsec behavior is from an environmental source, most notably a VPN gateway (e.g through receipt of required connection parameters from a VPN gateway), the evaluator shall ensure that the operational guidance contains any appropriate information for ensuring that this configuration can be properly applied.

Note in this case that the implementation of the IPsec protocol must be enforced entirely within the TOE boundary; i.e. it is not permissible for a software application TOE to be a graphical front-end for IPsec functionality implemented totally or in part by the underlying OS platform. The behavior referenced here is for the possibility that the configuration of the IPsec connection is initiated from outside the TOE, which is permissible so long as the TSF is solely responsible for enforcing the configured behavior. However, it is allowable for the TSF to rely on low-level platform-provided networking functions to implement the SPD from the client (e.g., enforcement of packet routing decisions).

Tests
As a prerequisite for performing the Test EAs for the individual FCS_IPSEC_EXT.1 elements below, the evaluator shall do the following:

The evaluator shall minimally create a test environment equivalent to the test environment illustrated below. It is expected that the traffic generator is used to construct network packets and will provide the evaluator with the ability manipulate fields in the ICMP, IPv4, IPv6, UDP, and TCP packet headers. The evaluator shall provide justification for any differences in the test environment.

Figure 2: Test Environment
Note that the evaluator shall perform all tests using the VPN client and a representative sample of platforms listed in the ST (for TOEs that claim to support multiple platforms).
FCS_IPSEC_EXT.1.1
TSS
The evaluator shall examine the TSS and determine that it describes how the IPsec capabilities are implemented.

If the TOE is a standalone software application, the evaluator shall ensure that the TSS asserts that all IPsec functionality is implemented by the TSF. The evaluator shall also ensure that the TSS identifies what platform functionality the TSF relies upon to support its IPsec implementation, if any (e.g. does it invoke cryptographic primitive functions from the platform's cryptographic library, enforcement of packet routing decisions by low-level network drivers).

If the TOE is part of a general-purpose desktop or mobile operating system, the evaluator shall ensure that the TSS describes at a high level the architectural relationship between the VPN client portion of the TOE and the rest of the TOE (e.g. is the VPN client an integrated part of the OS or is it a standalone executable that is bundled into the OS package). If the SPD is implemented by the underlying platform in this case, then the TSS describes how the client interacts with the platform to establish and populate the SPD, including the identification of the platform's interfaces that are used by the client.

In all cases, the evaluator shall also ensure that the TSS describes how the client interacts with the network stack of the platform(s) on which it can run (e.g., does the client insert itself within the stack via kernel mods, does the client simply invoke APIs to gain access to network services).

The evaluator shall ensure that the TSS describes how the SPD is implemented and the rules for processing both inbound and outbound packets in terms of the IPsec policy. The TSS describes the rules that are available and the resulting actions available after matching a rule. The TSS describes how the available rules and actions form the SPD using terms defined in RFC 4301 such as BYPASS (e.g., no encryption), DISCARD (e.g., drop the packet), and PROTECT (e.g., encrypt the packet) actions defined in RFC 4301. As noted in section 4.4.1 of RFC 4301, the processing of entries in the SPD is non-trivial and the evaluator shall determine that the description in the TSS is sufficient to determine which rules will be applied given the rule structure implemented by the TOE. For example, if the TOE allows specification of ranges, conditional rules, etc., the evaluator shall determine that the description of rule processing (for both inbound and outbound packets) is sufficient to determine the action that will be applied, especially in the case where two different rules may apply. This description shall cover both the initial packets (that is, no SA is established on the interface or for that particular packet) as well as packets that are part of an established SA.

Guidance

The evaluator shall examine the operational guidance to verify it describes how the SPD is created and configured. If there is an administrative interface to the client, then the guidance describes how the administrator specifies rules for processing a packet. The description includes all three cases - a rule that ensures packets are encrypted/decrypted, dropped, and allowing a packet to flow in plaintext. The evaluator shall determine that the description in the operational guidance is consistent with the description in the TSS, and that the level of detail in the operational guidance is sufficient to allow the administrator to set up the SPD in an unambiguous fashion. This includes a discussion of how ordering of rules impacts the processing of an IP packet.

If the client is configured by an external application, such as the VPN gateway, then the operational guidance should indicate this and provide a description of how the client is configured by the external application. The description should contain information as to how the SPD is established and set up in an unambiguous fashion. The description should also include what is configurable via the external application, how ordering of entries may be expressed, as well as the impacts that ordering of entries may have on the packet processing.

In either case, the evaluator ensures the description provided In the TSS is consistent with the capabilities and description provided in the operational guidance.

Tests

Depending on the implementation, the evaluator may be required to use a VPN gateway or some form of application to configure the client. For Test 2, the evaluator is required to choose an application that allows for the configuration of the full set of capabilities of the VPN client (in conjunction with the platform). For example, if the client provides a robust interface that allows for specification of wildcards, subnets, etc., it is unacceptable for the evaluator to choose a VPN Gateway that only allows for specifying a single fully qualified IP addresses in the rule.

The evaluator shall perform the following tests:

- **Test 1:** *: The evaluator shall configure an SPD on the client that is capable of the following: dropping a packet, encrypting a packet, and allowing a packet to flow in plaintext. The selectors used in the construction of the rule shall be different such that the evaluator can generate a packet and send packets to the client with the appropriate fields (fields that are used by the rule - e.g., the IP addresses, TCP/UDP ports) in the packet header. The evaluator performs both positive and negative test cases for each type of rule. The evaluator observes via the audit trail, and packet captures that the TOE exhibited the expected behavior: appropriate packets were dropped, allowed through without modification, was encrypted by the IPsec implementation.*
- **Test 2:** *The evaluator shall devise several tests that cover a variety of scenarios for packet processing. These scenarios must exercise the range of possibilities for SPD entries and processing modes as outlined in the TSS and operational guidance. Potential areas to cover include rules with overlapping ranges and conflicting entries, inbound and outbound packets, and packets that establish SAs as well as packets that belong to established SAs. The evaluator shall verify, via the audit trail and packet captures, for each scenario that the expected behavior is exhibited, and is consistent with both the TSS and the operational guidance.*

FCS_IPSEC_EXT.1.2

TSS

The evaluator shall check the TSS to ensure it states that the VPN can be established to operate in tunnel mode and/or transport mode (as selected).

Guidance

The evaluator shall confirm that the operational guidance contains instructions on how to configure the connection in each mode selected.

If both transport mode and tunnel mode are implemented, the evaluator shall review the operational guidance to determine how the use of a given mode is specified.

Tests
The evaluator shall perform the following test(s) based on the selections chosen:

- **Test 1:** *[conditional]: If tunnel mode is selected, the evaluator uses the operational guidance to configure the TOE to operate in tunnel mode and also configures a VPN gateway to operate in tunnel mode. The evaluator configures the TOE and the VPN gateway to use any of the allowable cryptographic algorithms, authentication methods, etc. to ensure an allowable SA can be negotiated. The evaluator shall then initiate a connection from the client to connect to the VPN GW peer. The evaluator observes (for example, in the audit trail and the captured packets) that a successful connection was established using the tunnel mode.*
- **Test 2:** *[conditional]: If transport mode is selected, the evaluator uses the operational guidance to configure the TOE to operate in transport mode and also configures an IPsec peer to accept IPsec connections using transport mode. The evaluator configures the TOE and the endpoint device to use any of the allowed cryptographic algorithms, authentication methods, etc. to ensure an allowable SA can be negotiated. The evaluator then initiates a connection from the TOE to connect to the remote endpoint. The evaluator observes (for example, in the audit trail and the captured packets) that a successful connection was established using the transport mode.*
- **Test 3:** *[conditional]: If both tunnel mode and transport mode are selected, the evaluator shall perform both Test 1 and Test 2 above, demonstrating that the TOE can be configured to support both modes.*
- **Test 4:** *[conditional]: If both tunnel mode and transport mode are selected, the evaluator shall modify the testing for FCS_IPSEC_EXT.1 to include the supported mode for SPD PROTECT entries to show that they only apply to traffic that is transmitted or received using the indicated mode.*

FCS_IPSEC_EXT.1.3
TSS
The evaluator shall examine the TSS to verify that the TSS provides a description of how a packet is processed against the SPD and that if no "rules" are found to match, that a final rule exists, either implicitly or explicitly, that causes the network packet to be discarded.

Guidance
The evaluator shall check that the operational guidance provides instructions on how to construct or acquire the SPD and uses the guidance to configure the TOE for the following test.

Tests
The evaluator shall perform the following test:

- **Test 1:** *The evaluator shall configure the SPD such that it has entries that contain operations that DISCARD, PROTECT, and (if applicable) BYPASS network packets. The evaluator may use the SPD that was created for verification of FCS_IPSEC_EXT.1.1. The evaluator shall construct a network packet that matches a BYPASS entry and send that packet. The evaluator should observe that the network packet is passed to the proper destination interface with no modification. The evaluator shall then modify a field in the packet header; such that it no longer matches the evaluator-created entries (there may be a "TOE-created" final entry that discards packets that do not match any previous entries). The evaluator sends the packet, and observes that the packet was not permitted to flow to any of the TOE's interfaces.*

FCS_IPSEC_EXT.1.4
TSS
The evaluator shall examine the TSS to verify that the algorithms AES-GCM-128 and AES-GCM-256 are implemented. If the ST author has selected either AES-CBC-128 or AES-CBC-256 in the requirement, then the evaluator verifies the TSS describes these as well. In addition, the evaluator ensures that the SHA- based HMAC algorithm conforms to the algorithms specified in the relevant iteration of FCS_COP.1 from the Base-PP that applies to keyed-hash message authentication.

Guidance
The evaluator checks the operational guidance to ensure it provides instructions on how the TOE is configured to use the algorithms selected in this component and whether this is performed through direct configuration, defined during initial installation, or defined by acquiring configuration settings from an environmental component.

Tests
- **Test 1:** *The evaluator shall configure the TOE as indicated in the operational guidance configuring the TOE to using each of the AES-GCM-128, and AES-GCM-256 algorithms, and attempt to establish a connection using ESP. If the ST Author has selected either AES-CBC-128 or AES-CBC-256, the TOE is configured to use those algorithms and the evaluator*

attempts to establish a connection using ESP for those algorithms selected.

FCS_IPSEC_EXT.1.5

TSS

The evaluator shall examine the TSS to verify that IKEv1 and/or IKEv2 are implemented. If IKEv1 is implemented, the evaluator shall verify that the TSS indicates whether or not XAUTH is supported, and that aggressive mode is not used for IKEv1 Phase 1 exchanges (i.e. only main mode is used). It may be that these are configurable options.

Guidance

The evaluator shall check the operational guidance to ensure it instructs the administrator how to configure the TOE to use IKEv1 and/or IKEv2 (as selected), and uses the guidance to configure the TOE to perform NAT traversal for the test below. If XAUTH is implemented, the evaluator shall verify that the operational guidance provides instructions on how it is enabled or disabled.

If the TOE supports IKEv1, the evaluator shall verify that the operational guidance either asserts that only main mode is used for Phase 1 exchanges, or provides instructions for disabling aggressive mode.

Tests

- **Test 1:** *The evaluator shall configure the TOE so that it will perform NAT traversal processing as described in the TSS and RFC 7296, section 2.23. The evaluator shall initiate an IPsec connection and determine that the NAT is successfully traversed. If the TOE supports IKEv1 with or without XAUTH, the evaluator shall verify that this test can be successfully repeated with XAUTH enabled and disabled in the manner specified by the operational guidance. If the TOE only supports IKEv1 with XAUTH, the evaluator shall verify that connections not using XAUTH are unsuccessful. If the TOE only supports IKEv1 without XAUTH, the evaluator shall verify that connections using XAUTH are unsuccessful.*
- **Test 2:** *[conditional]: If the TOE supports IKEv1, the evaluator shall perform any applicable operational guidance steps to disable the use of aggressive mode and then attempt to establish a connection using an IKEv1 Phase 1 connection in aggressive mode. This attempt should fail. The evaluator shall show that the TOE will reject a VPN gateway from initiating an IKEv1 Phase 1 connection in aggressive mode. The evaluator should then show that main mode exchanges are supported.*

FCS_IPSEC_EXT.1.6

TSS

The evaluator shall ensure the TSS identifies the algorithms used for encrypting the IKEv1 and/or IKEv2 payload, and that the algorithms AES-CBC-128, AES-CBC-256 are specified, and if others are chosen in the selection of the requirement, those are included in the TSS discussion.

Guidance

The evaluator checks the operational guidance to ensure it provides instructions on how the TOE is configured to use the algorithms selected in this component and whether this is performed through direct configuration, defined during initial installation, or defined by acquiring configuration settings from an environmental component.

Tests

The evaluator shall use the operational guidance to configure the TOE (or to configure the Operational Environment to have the TOE receive configuration) to perform the following test for each ciphersuite selected:

- **Test 1:** *The evaluator shall configure the TOE to use the ciphersuite under test to encrypt the IKEv1 and/or IKEv2 payload and establish a connection with a peer device, which is configured to only accept the payload encrypted using the indicated ciphersuite. The evaluator will confirm the algorithm was that used in the negotiation. The evaluator will confirm that the connection is successful by confirming that data can be passed through the connection once it is established. For example, the evaluator may connect to a webpage on the remote network and verify that it can be reached.*

FCS_IPSEC_EXT.1.7

TSS

There are no TSS EAs for this requirement.

Guidance

The evaluator shall check the operational guidance to ensure it provides instructions on how the TOE configures the values for SA lifetimes. In addition, the evaluator shall check that the guidance has the option for either the Administrator or VPN Gateway to configure Phase 1 SAs if time-based limits are supported. Currently there are no values mandated for the number of packets or number of bytes, the evaluator shall simply check the operational guidance to ensure

that this can be configured if selected in the requirement.

Tests
When testing this functionality, the evaluator needs to ensure that both sides are configured appropriately. From the RFC "A difference between IKEv1 and IKEv2 is that in IKEv1 SA lifetimes were negotiated. In IKEv2, each end of the SA is responsible for enforcing its own lifetime policy on the SA and rekeying the SA when necessary. If the two ends have different lifetime policies, the end with the shorter lifetime will end up always being the one to request the rekeying. If the two ends have the same lifetime policies, it is possible that both will initiate a rekeying at the same time (which will result in redundant SAs). To reduce the probability of this happening, the timing of rekeying requests SHOULD be jittered."

Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:

Each of the following tests shall be performed for each version of IKE selected in the FCS_IPSEC_EXT.1.5 protocol selection:

- ***Test 1:*** *[conditional]: The evaluator shall configure a maximum lifetime in terms of the # of packets (or bytes) allowed following the operational guidance. The evaluator shall establish an SA and determine that once the allowed # of packets (or bytes) through this SA is exceeded, the connection is closed.*
- ***Test 2:*** *[conditional]: The evaluator shall construct a test where a Phase 1 SA is established and attempted to be maintained for more than 24 hours before it is renegotiated. The evaluator shall observe that this SA is closed or renegotiated in 24 hours or less. If such an action requires that the TOE be configured in a specific way, the evaluator shall implement tests demonstrating that the configuration capability of the TOE works as documented in the operational guidance.*
- ***Test 3:*** *[conditional]: The evaluator shall perform a test similar to Test 2 for Phase 2 SAs, except that the lifetime will be 8 hours or less instead of 24 hours or less.*
- ***Test 4:*** *[conditional]: If a fixed limit for IKEv1 SAs is supported, the evaluator shall establish an SA and observe that the connection is closed after the fixed traffic and/or time value is reached.*

FCS_IPSEC_EXT.1.8
TSS
The evaluator shall check to ensure that the DH groups specified in the requirement are listed as being supported in the TSS. If there is more than one DH group supported, the evaluator checks to ensure the TSS describes how a particular DH group is specified/negotiated with a peer.

Guidance
There are no AGD EAs for this requirement.

Tests
The evaluator shall perform the following test:
- ***Test 1:*** *For each supported DH group, the evaluator shall test to ensure that all supported IKE protocols can be successfully completed using that particular DH group.*

FCS_IPSEC_EXT.1.9
TSS
The evaluator shall check to ensure that, for each DH group supported, the TSS describes the process for generating "x" (as defined in FCS_IPSEC_EXT.1.9) and each nonce. The evaluator shall verify that the TSS indicates that the random number generated that meets the requirements in this EP is used, and that the length of "x" and the nonces meet the stipulations in the requirement.

Guidance
There are no AGD EAs for this requirement.

Tests
There are no test EAs for this requirement.
FCS_IPSEC_EXT.1.10
EAs for this element are tested through EAs for FCS_IPSEC_EXT.1.9.
FCS_IPSEC_EXT.1.11
TSS
The evaluator ensures that the TSS identifies RSA and/or ECDSA as being used to perform peer authentication.

If pre-shared keys are chosen in the selection, the evaluator shall check to ensure that the TSS describes how pre-shared keys are established and used in authentication of IPsec connections. The description in the TSS shall also indicate how pre-shared key establishment is accomplished depending on whether the TSF can generate a pre-shared key, accept a pre-shared key, or both.

The evaluator shall ensure that the TSS describes how the TOE compares the peer's presented identifier to the reference identifier. This description shall include whether the certificate presented identifier is compared to the ID payload presented identifier, which field(s) of the certificate are used as the presented identifier (DN, Common Name, or SAN) and, if multiple fields are supported, the logical order comparison. If the ST author assigned an additional identifier type, the TSS description shall also include a description of that type and the method by which that type is compared to the peer's presented certificate.

Guidance
The evaluator shall check that the operational guidance describes how pre-shared keys are to be generated and established.

The evaluator ensures the operational guidance describes how to set up the TOE to use the cryptographic algorithms RSA and/or ECDSA.

In order to construct the environment and configure the TOE for the following tests, the evaluator will ensure that the operational guidance also describes how to configure the TOE to connect to a trusted CA, and ensure a valid certificate for that CA is loaded into the TOE as a trusted CA.

The evaluator shall also ensure that the operational guidance includes the configuration of the reference identifier(s) for the peer.

Tests
For efficiency's sake, the testing that is performed here has been combined with the testing for **FIA_X509_EXT.2** *and* **FIA_X509_EXT.3** *(for IPsec connections and depending on the Base-PP),* **FCS_IPSEC_EXT.1.12**, *and* **FCS_IPSEC_EXT.1.13**. *The following tests shall be repeated for each peer authentication protocol selected in the* **FCS_IPSEC_EXT.1.11** *selection above:*

- **Test 1:** *The evaluator shall have the TOE generate a public-private key pair, and submit a CSR (Certificate Signing Request) to a CA (trusted by both the TOE and the peer VPN used to establish a connection) for its signature. The values for the DN (Common Name, Organization, Organizational Unit, and Country) will also be passed in the request. Alternatively, the evaluator may import to the TOE a previously generated private key and corresponding certificate.*
- **Test 2:** *The evaluator shall configure the TOE to use a private key and associated certificate signed by a trusted CA and shall establish an IPsec connection with the peer.*
- **Test 3:** *: The evaluator shall test that the TOE can properly handle revoked certificates – conditional on whether CRL or OCSP is selected; if both are selected, and then a test is performed for each method. For this current version of the PP-Module, the evaluator has to only test one up in the trust chain (future drafts may require to ensure the validation is done up the entire chain). The evaluator shall ensure that a valid certificate is used, and that the SA is established. The evaluator then attempts the test with a certificate that will be revoked (for each method chosen in the selection) to ensure when the certificate is no longer valid that the TOE will not establish an SA*
- **Test 4:** *[conditional]: The evaluator shall generate a pre-shared key and use it, as indicated in the operational guidance, to establish an IPsec connection with the VPN GW peer. If the generation of the pre-shared key is supported, the evaluator shall ensure that establishment of the key is carried out for an instance of the TOE generating the key as well as an instance of the TOE merely taking in and using the key.*
  For each supported identifier type (excluding DNs), the evaluator shall repeat the following tests:
- **Test 5:** *For each field of the certificate supported for comparison, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the field in the peer's presented certificate and shall verify that the IKE authentication succeeds.*
- **Test 6:** *For each field of the certificate support for comparison, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to not match the field in the peer's presented certificate and shall verify that the IKE authentication fails.*
  The following tests are conditional:
- **Test 7:** *[conditional]: If, according to the TSS, the TOE supports both Common Name and SAN certificate fields and uses the preferred logic outlined in the Application Note, the tests above with the Common Name field shall be performed using peer certificates with no SAN extension. Additionally, the evaluator shall configure the peer's reference identifier on the TOE to not match the SAN in the peer's presented certificate but to match the Common Name in the peer's presented certificate, and verify that the IKE authentication fails.*
- **Test 8:** *[conditional]: If the TOE supports DN identifier types, the evaluator shall configure the peer's reference identifier on the TOE (per the administrative guidance) to match the subject DN in the peer's presented certificate and shall verify that the IKE authentication succeeds. To demonstrate a bit-wise comparison of the DN, the evaluator shall change a single bit in the DN (preferably, in an Object Identifier (OID) in the DN) and verify that the IKE authentication fails.* **To demonstrate a comparison of DN values, the evaluator shall change any one of the four DN values and verify that the IKE authentication fails.**

- **Test 9:** *[conditional]: If the TOE supports both IPv4 and IPv6 and supports IP address identifier types, the evaluator must repeat test 1 and 2 with both IPv4 address identifiers and IPv6 identifiers. Additionally, the evaluator shall verify that the TOE verifies that the IP header matches the identifiers by setting the presented identifiers and the reference identifier with the same IP address that differs from the actual IP address of the peer in the IP headers and verifying that the IKE authentication fails.*
- **Test 10:** *[conditional]: If, according to the TSS, the TOE performs comparisons between the peer's ID payload and the peer's certificate, the evaluator shall repeat the following test for each combination of supported identifier types and supported certificate fields (as above). The evaluator shall configure the peer to present a different ID payload than the field in the peer's presented certificate and verify that the TOE fails to authenticate the IKE peer.*

FCS_IPSEC_EXT.1.12
EAs for this element are tested through EAs for FCS_IPSEC_EXT.1.11.
FCS_IPSEC_EXT.1.13
EAs for this element are tested through EAs for FCS_IPSEC_EXT.1.11.
FCS_IPSEC_EXT.1.14
TSS
The evaluator shall check that the TSS describes the potential strengths (in terms of the number of bits in the symmetric key) of the algorithms that are allowed for the IKE and ESP exchanges. The TSS shall also describe the checks that are done when negotiating IKEv1 Phase 2 and/or IKEv2 CHILD_SA suites to ensure that the strength (in terms of the number of bits of key in the symmetric algorithm) of the negotiated algorithm is less than or equal to that of the IKE SA this is protecting the negotiation.

Guidance
There are no AGD EAs for this requirement.

Tests
The evaluator follows the guidance to configure the TOE to perform the following tests:
- **Test 1:** *This test shall be performed for each version of IKE supported. The evaluator shall successfully negotiate an IPsec connection using each of the supported algorithms and hash functions identified in the requirements.*
- **Test 2:** *[conditional]: This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an SA for ESP that selects an encryption algorithm with more strength than that being used for the IKE SA (i.e., symmetric algorithm with a key size larger than that being used for the IKE SA). Such attempts should fail.*
- **Test 3:** *This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an IKE SA using an algorithm that is not one of the supported algorithms and hash functions identified in the requirements. Such an attempt should fail.*
- **Test 4:** *This test shall be performed for each version of IKE supported. The evaluator shall attempt to establish an SA for ESP (assumes the proper parameters where used to establish the IKE SA) that selects an encryption algorithm that is not identified in FCS_IPSEC_EXT.1.4. Such an attempt should fail.*

5.2.3 User Data Protection (FDP)

FDP_RIP.2 Full Residual Information Protection

FDP_RIP.2.1

The **[selection: *TOE*, *TOE platform*]** shall enforce that any previous information content of a resource is made unavailable upon the [**selection**: *allocation of the resource to, deallocation of the resource from*] all objects.

Application Note: This requirement ensures, for example, that protocol data units (PDUs) are not padded with residual information such as cryptographic key material. The ST author uses the selection to specify when previous information is made unavailable.

Evaluation Activities ▽

FDP_RIP.2
TSS
Requirement met by the platform

The evaluator shall examine the TSS to verify that it describes (for each supported platform) the extent to which the client processes network packets and addresses the FDP_RIP.2 requirement.

Requirement met by the TOE

"Resources" in the context of this requirement are network packets being sent through (as opposed to "to", as is the case when a security administrator connects to the TOE) the TOE. The concern is that once a network packet is sent, the buffer or memory area used by the packet still contains data from that packet, and that if that buffer is re-used, those data might remain and make their way into a new packet. The evaluator shall check to ensure that the TSS describes packet processing to the extent that they can determine that no data will be reused when processing network packets. The evaluator shall ensure that this description at a minimum describes how the previous data are zeroized/overwritten, and at what point in the buffer processing this occurs.
Guidance
There are no AGD EAs for this requirement.

Tests
There are no test EAs for this requirement.

5.2.4 Security Management (FMT)

The TOE is not required to maintain a separate management role. It is, however, required to provide functionality to configure certain aspects of TOE operation that should not be available to the general user population. It is possible for the TOE, TOE Platform, or VPN Gateway to provide this functionality. The client itself has to be configurable - whether it is from the EUD or from a VPN gateway.

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

FMT_SMF.1.1/VPN

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

- *Specify VPN gateways to use for connections,*
- *Specify IPsec VPN Clients to use for connections,*
- *Specify IPsec-capable network devices to use for connections,*
- *Specify client credentials to be used for connections,*
- *Configure the reference identifier of the peer,*
- *[**assignment**: any additional management functions]*

]

Application Note: Several of the management functions defined above correspond to the use cases of the TOE as follows:

- "Specify VPN gateways to use for connections" – Use Case 1
- "Specify IPsec VPN Clients to use for connections" – Use Case 2 (specifically refers to different end points to use for client-to-client connections)
- "Specify IPsec-capable network devices to use for connections" – Use Case 3

Selections appropriate for the use case(s) supported by the TOE should be claimed. "Client credentials" will include the client certificate used for IPsec authentication, and may also include a username/password.

For TOEs that support only IP address and FQDN identifier types, configuration of the reference identifier may be the same as configuration of the peer's name for the purposes of connection.

If there are additional management functions performed by the TOE (including those specified in FCS_IPSEC_EXT.1), they should be added in the assignment.

Evaluation Activities ▼

FMT_SMF.1/VPN
TSS
The evaluator shall check to ensure the TSS describes the client credentials and how they are used by the TOE.

Guidance
The evaluator shall check to make sure that every management function mandated in the ST for this requirement is described in the operational guidance and that the description contains the information required to perform the management duties associated with each management function.

Tests
The evaluator shall test the TOE's ability to provide the management functions by configuring the TOE according to the operational guidance and testing each management activity listed in the ST.

The evaluator shall ensure that all management functions claimed in the ST can be performed by

completing activities described in the AGD. Note that this may be performed in the course of completing other testing.

5.2.5 Protection of the TSF (FPT)

FPT_TST_EXT.1/VPN TSF Self-Test

FPT_TST_EXT.1.1/VPN

> The [**selection**: *TOE, TOE platform*] shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2/VPN

> The [**selection**: *TOE, TOE platform*] shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the [**assignment**: *cryptographic services provided either by the portion of the TOE described by the Base-PP or by the operational environment*].

> **Application Note:** While the TOE is typically a software package running in the IT Environment, it is still capable of performing the self-test activities required above. It should be understood, however, that there is a significant dependency on the host environment in assessing the assurance provided by the tests mentioned above (meaning that if the host environment is compromised, the self-tests will not be meaningful).

> Cryptographic verification of the integrity is required, but the method by which this can be accomplished is specified in the ST in the assignment. The ST author will fill in the assignment with references to the cryptographic functions used to perform the integrity checks; this will include hashing and may potentially include digital signatures signed using X.509 certificates. If the TSF provides the cryptographic services used to verify updates, all relevant FCS_COP requirements will be identified in the assignment by the ST author.

Evaluation Activities ▼

FPT_TST_EXT.1/VPN
Except for where it is explicitly noted, the evaluator is expected to check the following information regardless of whether the functionality is implemented by the TOE or by the TOE platform.

TSS
The evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF on startup; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" shall be used). The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly. If some of the tests are performed by the TOE platform, the evaluator shall check the TSS to ensure that those tests are identified, and that the ST for each platform contains a description of those tests. Note that the tests that are required by this component are those that support security functionality in the VPN Client PP-Module, which may not correspond to the set of all self-tests contained in the platform STs.

The evaluator shall examine the TSS to ensure that it describes how the integrity of stored TSF executable code is cryptographically verified when it is loaded for execution. The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the integrity of stored TSF executable code has not been compromised. The evaluator shall check to ensure that the cryptographic requirements listed are consistent with the description of the integrity verification process.

The evaluator also ensures that the TSS (or the operational guidance) describes the actions that take place for successful (e.g. hash verified) and unsuccessful (e.g., hash not verified) cases. For checks implemented entirely by the platform, the evaluator ensures that the operational guidance for the TOE references or includes the platform-specific guidance for each platform listed in the ST.

Guidance
If not present in the TSS, the evaluator ensures that the operational guidance describes the actions that take place for successful (e.g. hash verified) and unsuccessful (e.g., hash not verified) cases. For checks implemented entirely by the platform, the evaluator ensures that the operational guidance for the TOE references or includes the platform-specific guidance for each platform listed in the ST.

Tests
The evaluator shall perform the following tests:
- ***Test 1:*** *The evaluator performs the integrity check on a known good TSF executable and*

> *verifies that the check is successful.*
> - **Test 2:** *The evaluator modifies the TSF executable, performs the integrity check on the modified TSF executable and verifies that the check fails.*

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
FAU_STG.4	This SFR supports the objective by requiring the TSF to disable the execution of auditable events if the audit trail cannot be written to.	
FAU_STG.1 (from Base-PP)	This SFR supports the objective by ensuring that stored audit records are protected against unauthorized modification or destruction.	
FAU_SAR.1 (optional)	This SFR supports the objective by ensuring that stored audit records are protected against unauthorized access.	
FIA_X509_EXT.1/Rev (from Base-PP)	This SFR supports the objective by defining the TOE functionality for certificate validation.	
FIA_X509_EXT.3 (from Base-PP)	This SFR supports the objective by defining the mechanism by which the TOE generates certificate signing requests, which includes validation of the certificate provided in response.	
FDP_CER_EXT.1	This SFR supports the objective by defining the rules the TOE must use to generate and issue proxy TLS server certificates from its internal CA.	
FDP_CER_EXT.2	This SFR supports the objective by requiring the TOE to link the certificates presented for TLS connectivity with the certificates it issues from its internal CA.	
FDP_CER_EXT.3	This SFR supports the objective by defining the rules for the TOE's issuing of proxy TLS server certificates.	
FDP_CSIR_EXT.1	This SFR supports the objective by defining how the TOE can ensure the use of fresh certificates.	
FIA_ENR_EXT.1	This SFR supports the objective by defining the mechanism by which the TOE requests a certificate for its own embedded CA's signing key.	
FIA_X509_EXT.1/STIP	This SFR supports the objective by defining the certificate validation rules that must be followed for certificates that are used for proxy TLS connections.	
FIA_X509_EXT.2/STIP	This SFR supports the objective by defining the certificate authentication behavior for STIP connections.	
FDP_PIN_EXT.1 (optional)	This SFR supports the objective by defining the optional implementation of certificate pinning.	
FIA_ESTC_EXT.2 (optional)	This SFR supports the objective by defining requirements for the composition of EST requests if the TOE supports EST.	
FDP_CER_EXT.4 (selection-based)	This SFR supports the objective by defining the rules the TOE must use to generate and issue proxy TLS client certificates from its internal CA if mutual authentication is supported.	
FDP_CER_EXT.5 (selection-based)	This SFR supports the objective by defining the rules for the TOE's issuing of proxy TLS client certificates if mutual authentication is supported.	
FDP_CRL_EXT.1 (selection-based)	This SFR supports the objective by defining rules for the generation of CRLs if the TOE uses this as the mechanism to ensure the freshness of its issued certificates.	
FDP_CSI_EXT.1 (selection-based)	This SFR supports the objective by defining the revocation echecking method supported by the TOE for the proxy TLS server certificates it issues, if revocation is how the freshness of its issued certificates is assured.	

FDP_CSI_EXT.2 (selection-based)	This SFR supports the objective by defining the revocation echecking method supported by the TOE for the proxy TLS client certificates it issues, if mutual authentication is supported and revocation is how the freshness of its issued certificates is assured.
FDP_OCSP_EXT.1 (selection-based)	This SFR supports the objective by defining rules for the generation of OCSP responses if the TOE uses this as the mechanism to ensure the freshness of its issued certificates.
FDP_OCSPS_EXT.1 (selection-based)	This SFR supports the objective by defining rules for the implementation of OCSP stapling if the TOE supports this functionality.
FIA_ESTC_EXT.1 (selection-based)	This SFR supports the objective by defining requirements for the implementation of EST if the TOE uses this mechanism to obtain TLS certificates for its own use.
FTA_TAB.1 (from Base-PP	This SFR supports the objective by applying a warning banner to any interface used by an administrator to access the TOE.
FTA_TAB.1/TLS (selection-based)	This SFR supports the objective by optionally applying a warning banner to a user whose network activity passes through the TOE for decryption and potential inspection.
FCS_STG_EXT.1	This SFR supports the objective by requiring the TOE to implement hardware-based protection for stored keys.
FDP_STG_EXT.1	This SFR supports the objective by defining the mechanism used to protect public key data from unauthorized modification.
FPT_KST_EXT.1	This SFR supports the objective by requiring the TSF to enforce the prevention of plaintext key export.
FPT_KST_EXT.2	This SFR supports the objective by preventing the unauthorized use of secret and private keys.
FCS_CKM_EXT.5 (selection-based)	This SFR supports the objective by defining the integrity mechanism used to guarantee the integrity of public key data.
FCS_CKM.4 (from Base-PP)	This SFR supports the objective by ensuring secret and private key data is disposed of immediately after use to prevent unauthorized disclosure of keys.
FCS_TLSC_EXT.1 (from Base-PP)	This SFR supports the objective by defining requirements for the TOE's implementation of TLS as a client.
FCS_TLSS_EXT.1 (from Base-PP)	This SFR supports the objective by defining requirements for the TOE's implementation of TLS as a server.
FTP_ITC.1 (refined from Base-PP)	This SFR supports the objective by defining the TOE interfaces that require protected communications as well as the methods of protection applied to these interfaces.
FCS_COP.1/STIP	This SFR supports the objective by defining cryptographic algorithms the TOE must support for decryption and re-encryption of proxy TLS traffic.
FCS_TTTC_EXT.1	This SFR supports the objective by defining requirements for the TOE's implementation of TLS as a client, specifically in the case where the TOE is establishing a proxy connection between itself and the original requested TLS server.
FCS_TTTC_EXT.5	This SFR supports the objective by defining the Supported Groups used by the TOE's proxy TLS client interface.
FCS_TTTS_EXT.1	This SFR supports the objective by defining requirements for the TOE's implementation of TLS as a server, specifically in the case where the TOE is establishing a proxy connection between itself and the original monitored TLS client.
FDP_PPP_EXT.1	This SFR supports the objective by defining the processing rules that the TOE applies to plaintext traffic once decrypted.
FDP_PRC_EXT.1	This SFR supports the objective by defining requirements for the routing of decrypted plaintext traffic.
FDP_STIP_EXT.1	This SFR supports the objective by defining the TOE's ability to establish proxy TLS sessions between a monitored client and a

	requested server and to apply appropriate rules to the handling of the decrypted traffic.
FDP_TEP_EXT.1	This SFR supports the objective by defining the TOE's abbility to enforce filtering rules on TLS traffic passing through the TOE.
FCS_TTTC_EXT.3 (selection-based)	This SFR supports the objective by defining optional support for TLS mutual authentication that is applied to the TOE's proxy TLS client interface.
FCS_TTTC_EXT.4 (selection-based)	This SFR supports the objective by defining optional support for TLS session renegotiation that is applied to the TOE's proxy TLS client interface.
FCS_TTTS_EXT.3 (selection-based)	This SFR supports the objective by defining optional support for TLS mutual authentication that is applied to the TOE's proxy TLS server interface.
FCS_TTTS_EXT.4 (selection-based)	This SFR supports the objective by defining optional support for TLS session renegotiation that is applied to the TOE's proxy TLS server interface.
FDP_STIP_EXT.2 (selection-based)	This SFR supports the objective by defining the optional capability of the TOE to establish a proxy TLS session in the case where mutual authentication is supported.
FPT_FLS.1	This SFR supports the objective by requiring the TSF to preserve a secure state when certain failures occur.
FPT_RCV.1	This SFR supports the objective by requiring the TSF to support a maintenance mode of operation that is entered when certain failures occur.
FDP_RIP.1	This SFR supports the objective by defining the residual data that is cleared from TOE memory and when the clearing occurs.
FAU_STG_EXT.1 (from Base-PP)	This SFR supports the objective by defining a mechanism for the secure storage of audit data in the OE.
FAU_GEN.1/STIP	This SFR supports the objective by defining the auditable events specific to STIP functionality that the TSF must generate.
FAU_GCR_EXT.1	This SFR supports the objective by defining the mechanism the TOE uses to store certificate data.
FAU_SAR.3 (optional)	This SFR supports the objective by optionally defining the functionality to search audit records for events associated with a particular certificate.
FAU_SCR_EXT.1 (selection-based)	This SFR supports the objective by requiring the TOE to implement a search function for certificate storage if the TSF implements its own certificate store (as opposed to relying on environmental storage).
FMT_MOF.1	This SFR supports the objective by defining the authorized use of the TOE by association between the supported management functions and the roles that are authorized to perform them.
FMT_SMF.1/STIP	This SFR supports the objective by defining the TOE's management functions that are specific to STIP functionality.
FDP_SMR.2/STIP	This SFR supports the objective by defining additional management roles that the TOE may support that are specific to STIP 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 General Purpose Operating Systems PP 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 Protection Profile for General Purpose Operating Systems

6.1.1 Consistency of TOE Type

If this PP-Module is used to extend the NDcPP, the TOE type for the overall TOE is still a generic network device. However, one of the functions of the device must be the ability for it to act as an SSL/TLS Inspection Proxy. The TOE boundary is simply extended to include that functionality.

6.1.2 Consistency of Security Problem Definition

The threats, assumptions, and 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.UNTRUSTED_COMMUNICATION	The threat of untrusted communication can provide unauthorized access to unintended resources if using weak cryptography or use untrusted intermediate systems. This can be mitigated either by protocols defined in this PP-Module or in the Base-PP.
T.AUDIT	Auditing poses a threat if certain activities aren't logged, like the issuance of certificates. This threat can be mitigated if proper configurations are in place to prevent the compromise of audit data defined in this PP-Module or the Base-PP.
T.UNAUTHORIZED_USERS	The threat of unauthorized users attempting to gain access to other users' credentials can be addressed by placing protections for logged-in users and only allow privileged user access methods defined in this PP-Module or in the Base-PP.
T.CREDENTIALS	Beyond the Base-PP, the threat of manipulation of the CA signing key can be mitigated by providing access protection to persistent keys.
T.SERVICES	The threat of misuse or manipulation of services is not defined in the Base-PP, but it is consistent with the general threat of unauthorized manipulation of the TSF.
T.DEVICE_FAILURE	The failure of the certificate authority or routing traffic to inspection poses a threat not defined in the Base-PP.
T.UNAUTHORIZED_DISCLOSURE	The Base-PP does not include the threat of unauthorized disclosure to sensitive data that is only intended for the monitored client because this is an interface that the Base-PP cannot assume all conformant TOEs have.
T.INAPPROPRIATE_ACCESS	The threat of inappropriate access to unintended servers could disclose unauthorized traffic to inspection processes which is not defined in the Base-PP because a generic network device does not necessarily have a traffic inspection functionality.
P.AUTHORIZATION_TO_INSPECT	The Base-PP cannot define the interactions that an end user will have with a generic device because it may vary depending on the specific device type. This PP-Module defines a policy that is specific to the use case of a STIP device.

6.1.3 Consistency of Objectives

The objectives for the TOEs are consistent with the General Purpose Operating Systems PP based on the following rationale:

PP-Module TOE Objective	Consistency Rationale
O.AUDIT_LOSS_RESPONSE	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.AUDIT_PROTECTION	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.CERTIFICATES	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.DISPLAY_BANNER	The Base-PP does not define any TOE objectives so PP-Module

	objectives do not conflict with it.
O.PERSISTENT_KEY_PROTECTION	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.PROTECTED_COMMUNICATIONS	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.RECOVERY	The Base-PP does not define any TOE objectives so PP-Module objectives do not conflict with it.
O.RESIDUAL_INFORMATION_CLEARING	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 Operational Environment are consistent with the General Purpose Operating Systems PP based on the following rationale:

PP-Module Operational Environment Objective	Consistency Rationale
OE.AUDIT	This objective intends for the TOE's OE to have adequate storage to retain the TOE's audit records. This objective is not defined in the Base-PP but can be assumed to be consistent with the Base-PP because FAU_STG_EXT.1 requires transmission of audit data to an environmental audit server, which means that there should be some assurance of the security of that server.
OE.CERT_REPOSITORY	This objective intends for the TOE's OE to provide a certificate repository. This is not defined in the Base-PP because not all network devices will necessarily need to interface with a certificate repository.
OE.CERT_REPOSITORY_SEARCH	This objective intends for the TOE's OE which will provide a certificate repository to also have the capability to search within the repository. This is not defined in the Base-PP because not all network devices will necessarily need to interface with a certificate repository.

6.1.4 Consistency of Requirements

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

PP-Module Requirement	Consistency Rationale
	Modified SFRs
FAU_STG.1	Other than this SFR becoming mandatory versus optional, there is no modification to this SFR.
FCS_CKM.4	The ST author is instructed to include security critical parameters and when key destruction is required.
FCS_TLSC_EXT.1	Other than this SFR becoming mandatory versus selection-based, there is no modification to this SFR.
FCS_TLSS_EXT.1	Other than this SFR becoming mandatory versus selection-based, there is no modification to this SFR.
FIA_X509_EXT.1/Rev	Other than this SFR becoming mandatory versus selection-based, there is no modification to this SFR.
FIA_X509_EXT.2	The PP-Module partially completes selections in this SFR using the available options to specify minimum required functionality for X.509 authentication based on its use in STIP. The PP-Module also refines the authorized management roles that can perform the function defined in FIA_X509_EXT.2.2.
FIA_X509_EXT.3	There is no change to this SFR. Only its trigger for inclusion is changed because this

	PP-Module introduces an alternate method of obtaining a certificate for the TOE.
FTP_ITC.1	The PP-Module partially completes selections and assignments in this SFR using the available options to specify external interfaces and trusted channels that all STIP products must support at minimum.

Additional SFRs

This PP-Module does not add any requirements when the General Purpose Operating Systems PP is the base.

Mandatory SFRs

FCS_CKM.1/VPN	Generation of IKE peer authentication keys is added functionality that does not prevent the existing GPOS functions from being performed.
FCS_IPSEC_EXT.1	This SFR defines the VPN client's IPsec implementation, which is added functionality that does not interfere with the GPOS functions.
FDP_RIP.2	The requirement to protect against re-use of residual data is a property of the VPN client behavior and does not impact the GPOS functionality.
FMT_SMF.1/VPN	The ability to configure the VPN client behavior does not affect whether the GPOS as a whole can perform its security functions.
FPT_TST_EXT.1/VPN	Self-testing of the VPN client functionality does not impact the ability of the GPOS to perform its security functions.

Optional SFRs

This PP-Module does not define any Optional requirements.

Selection-based SFRs

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.

Objective SFRs

FAU_GEN.1/VPN	Audit records generated by the VPN client do not interfere with GPOS functionality. The possibility of the underlying OS platform generating audit records is consistent with the GPOS PP, which already contains FAU_GEN.1.
FAU_SEL.1/VPN	The ability to suppress the generation of certain audit records related to VPN activity does not interfere with the ability of the GPOS to satisfy its security functionality.
FDP_IFC_EXT.1/VPN	The ability of the VPN client to prevent split tunneling of IPsec traffic requires it to have hooks into lower-level OS behavior, but there are no requirements in the GPOS PP that would prevent this functionality from being supported.

Implementation-Dependent SFRs

This PP-Module does not define any Implementation-Dependent requirements.

6.2 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 GPOS PP, MDF PP, App PP, or MDM PP as well. These PPs include 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.

Appendix A - Optional SFRs

A.1 Strictly Optional Requirements

This PP-Module does not define any Strictly Optional SFRs.

A.2 Objective Requirements

A.2.1 Auditable Events for Objective SFRs

Table 4: Auditable Events for Objective Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FAU_GEN.1/VPN	No events specified	
FAU_SEL.1/VPN	All modifications to the audit configuration that occur while the audit collection functions are operating.	
FDP_IFC_EXT.1/VPN	No events specified	

A.2.2 Security Audit (FAU)

FAU_GEN.1/VPN Audit Data Generation

FAU_GEN.1.1/VPN

The TSF **and [selection: *TOE platform, no other component*]** shall be able to generate an audit record of the following auditable events:

a. Start-up and shutdown of the audit functions;
b. All auditable events for the [*not specified*] level of audit;
c. All administrative actions;
d. [*Specifically defined auditable events listed in **the Auditable Events tables***].

Application Note: In the case of "a", the audit functions referred to are those provided by the TOE. For example, in the case that the TOE was a stand-alone executable, auditing the startup and the shutdown of the TOE itself would be sufficient to meet the requirements of this clause.

Many auditable aspects of the SFRs included in this document deal with administrative actions. Item c above requires all administrative actions to be auditable, so no additional specification of the auditability of these actions is present in the Auditable Events table. While the TOE itself does not need to provide the ability to perform I&A for an administrator, this requirement implies that the TOE possess the capability to audit the events described by the Base-PP as "administrative actions" (primarily dealing with configuration of the functionality provided by the TOE).

The auditable events defined in the Auditable Events table are for the SFRs that are explicitly defined in this PP-Module. For any SFRs that are included as part of the TOE based on the claimed Base-PP, it is expected that any applicable auditable events defined for those SFRs in the Base-PP are also claimed as part of the TSF. These auditable events only apply if the client actually performs these functions. If the platform performs any of these actions, then the platform is responsible for performing the auditing, not the TSF

FAU_GEN.1.2/VPN

The TSF **and [selection: *TOE platform, no other component*]** shall record within each audit record at least the following information:

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

Evaluation Activities ⋁

FAU_GEN.1/VPN
TSS
The evaluator shall examine the TSS to determine that it describes the auditable events and the component that is responsible for each type of auditable event.

Guidance
The evaluator shall check the operational guidance and ensure that it lists all of the auditable events and provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator shall check to make sure that every audit event type mandated by the VPN Client PP-Module is described and that the description of the fields contains the information required in FAU_GEN.1.2/VPN, and the additional information specified in the Auditable Events table of the VPN Client PP-PP-Module.

In particular, the evaluator shall ensure that the operational guidance is clear in relation to the contents for failed cryptographic events. In the Auditable Events table of the VPN Client PP-Module, information detailing the cryptographic mode of operation and a name or identifier for the object being encrypted is required. The evaluator shall ensure that name or identifier is sufficient to allow an administrator reviewing the audit log to determine the context of the cryptographic operation (for example, performed during a key negotiation exchange, performed when encrypting data for transit) as well as the non-TOE endpoint of the connection for cryptographic failures relating to communications with other IT systems.

The evaluator shall also make a determination of the administrative actions that are relevant in the context of the VPN Client PP-Module. The TOE may contain functionality that is not evaluated in the context of the VPN Client PP-Module because the functionality is not specified in an SFR. This functionality may have administrative aspects that are described in the operational guidance. Since such administrative actions will not be performed in an evaluated configuration of the TOE, the evaluator shall examine the operational guidance and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the VPN Client PP-Module, which thus form the set of "all administrative actions". The evaluator may perform this activity as part of the activities associated with ensuring the AGD_OPE guidance satisfies the requirements.

For each required auditable event, the evaluator shall examine the operational guidance to determine that it is clear to the reader where each event is generated (e.g. the TSF may generate its own audit logs in one location while the platform-provided auditable events are generated elsewhere).

Tests
The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records in accordance with the EAs associated with the functional requirements in the VPN Client PP-Module. Additionally, the evaluator shall test that each administrative action applicable in the context of the VPN Client PP-Module is auditable. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries.

Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to ensure that the administrative guidance provided is correct verifies that AGD_OPE.1 is satisfied and should address the invocation of the administrative actions that are needed to verify the audit records are generated as expected.

FAU_SEL.1/VPN Selective Audit

FAU_SEL.1.1/VPN

The **[selection: *TSF, TOE platform*]** shall be able to select the set of events to be audited from the set of all auditable events based on the following attributes: [*event type, [success of auditable security events, failure of auditable security events], [**assignment***: list of additional attributes that audit selectivity is based upon]*].

Application Note: The intent of this requirement is to identify all criteria that can be selected to trigger an audit event. This can be configured through an interface on the client for a user/administrator to invoke, or it could be an interface that the VPN gateway uses to instruct the client on which events are to be audited. For the ST author, the assignment is used to list any additional criteria or "none". The auditable event types are listed in the Auditable Events table

The intent of the first selection is to allow for the case where the underlying platform is responsible for some audit log generation functionality.

Evaluation Activities ▼

TSS
There are no TSS EAs for this SFR.

Guidance
The evaluator shall review the administrative guidance to ensure that the guidance itemizes all event types, as well as describes all attributes that are to be selectable in accordance with the requirement, to include those attributes listed in the assignment. The administrative guidance shall also contain instructions on how to set the pre-selection, or how the VPN gateway will configure the client, as well as explain the syntax (if present) for multi-value pre-selection. The administrative guidance shall also identify those audit records that are always recorded, regardless of the selection criteria currently being enforced.

Tests
The evaluator shall perform the following tests:

- **Test 1:** *For each attribute listed in the requirement, the evaluator shall devise a test to show that selecting the attribute causes only audit events with that attribute (or those that are always recorded, as identified in the administrative guidance) to be recorded.*
- **Test 2:** *[conditional] If the TSF supports specification of more complex audit pre-selection criteria (e.g., multiple attributes, logical expressions using attributes) then the evaluator shall devise tests showing that this capability is correctly implemented. The evaluator shall also, in the test plan, provide a short narrative justifying the set of tests as representative and sufficient to exercise the capability.*

A.2.3 User Data Protection (FDP)

FDP_IFC_EXT.1/VPN Subset Information Flow Control

FDP_IFC_EXT.1.1/VPN

The TSF shall ensure that all IP traffic (other than IP traffic required to establish the VPN connection) flow through the IPsec VPN client.

Application Note: This requirement is mandatory when the MDF is the base PP (see FDP_IFC_EXT.1/ALL). Otherwise it is optional.

This requirement is used when the VPN client is able to enforce the requirement through its own components. This generally will have to be done through using hooks provided by the platform such that the TOE is able to ensure that no IP traffic can flow through other network interfaces.

Evaluation Activities ▼

TSS
The evaluator shall verify that the TSS section of the ST describes the routing of IP traffic through processes on the TSF when a VPN client is enabled. The evaluator shall ensure that the description indicates which traffic does not go through the VPN and which traffic does and that a configuration exists for each baseband protocol in which only the traffic identified by the ST author is necessary for establishing the VPN connection (IKE traffic and perhaps HTTPS or DNS traffic) is not encapsulated by the VPN protocol (IPsec). The ST author shall also identify in the TSS section any differences in the routing of IP traffic when using any supported baseband protocols (e.g. WiFi or, LTE).

Guidance
The evaluator shall verify that the following is addressed by the documentation:

- *The description above indicates that if a VPN client is enabled, all configurations route all IP traffic (other than IP traffic required to establish the VPN connection) through the VPN client.*
- *The AGD guidance describes how the user and/or administrator can configure the TSF to meet this requirement.*

Tests
The evaluator shall perform the following test:

Step 1 - The evaluator shall use the platform to enable a network connection without using IPsec. The evaluator shall use a packet sniffing tool between the platform and an Internet-connected network. The evaluator shall turn on the sniffing tool and perform actions with the device such as navigating to websites, using provided applications, accessing other Internet resources (Use Case 1), accessing another VPN client (Use Case 2), or accessing an IPsec-capable network device (Use Case 3). The evaluator shall verify that the sniffing tool captures the traffic generated by these actions, turn off the sniffing tool, and save the session data.

Step 2 - The evaluator shall configure an IPsec VPN client that supports the routing specified in this requirement, and if necessary, configure the device to perform the routing specified as

described in the AGD guidance. The evaluator shall turn on the sniffing tool, establish the VPN connection, and perform the same actions with the device as performed in the first step. The evaluator shall verify that the sniffing tool captures traffic generated by these actions, turn off the sniffing tool, and save the session data.

Step 3 - The evaluator shall examine the traffic from both step one and step two to verify that all IP traffic, aside from and after traffic necessary for establishing the VPN (such as IKE, DNS, and possibly HTTPS), is encapsulated by IPsec.

Step 4 - The evaluator shall attempt to send packets to the TOE outside the VPN connection and shall verify that the TOE discards them.

A.3 Implementation-Based Requirements

This PP-Module does not define any Implementation-Based SFRs.

Appendix B - Selection-Based Requirements

B.1 Auditable Events for Selection-based SFRs

Table 5: Auditable Events for Selection-based Requirements

Requirement	Auditable Events	Additional Audit Record Contents
FIA_PSK_EXT.1	No events specified	

B.2 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. There are two types of pre-shared keys that must be supported by the TOE, as specified in the requirements below. The first type is referred to as "text-based pre-shared keys," which refer to pre-shared keys that are entered by users as a string of characters from a standard character set, similar to a password. Such pre-shared keys must be conditioned so that the string of characters is transformed into a string of bits, which is then used as the key.

The second type is referred to as "bit-based pre-shared keys" (for lack of a standard term); this refers to keys that are either generated by the TSF on a command from the administrator, or input in "direct form" by an administrator. "Direct form" means that the input is used directly as the key, with no "conditioning" as was the case for text-based pre-shared keys. An example would be a string of hex digits that represent the bits that comprise the key.

The requirements below mandate that the TOE support text-based pre-shared keys. Bit-based preshared keys may or may not be supported, and if they are, generation of these keys may be done either by the TOE or in the operational environment.

FIA_PSK_EXT.1 Pre-Shared Key Composition

> ***The inclusion of this selection-based component depends upon selection in FCS_IPSEC_EXT.1.11.***

FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for IPsec.

FIA_PSK_EXT.1.2

The TSF shall be able to accept text-based pre-shared keys that:
- Are 22 characters and [**selection**: [***assignment**: other supported lengths]*, *no other lengths]*,
- Composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", and [**selection**: *no other special characters*, [***assignment**: list of additional supported special characters]]*.

FIA_PSK_EXT.1.3

The TSF shall condition the text-based pre-shared keys by using [**selection**: *SHA-1, SHA-256, SHA-512,* [***assignment**: method of conditioning text string]]*, [**selection**:
- *be able to* [**selection**: *accept, generate using the random bit generator specified in FCS_RBG_EXT.1]*,
- *perform no other conditioning*

].

Application Note: This SFR is claimed if "pre-shared keys" is selected in FCS_IPSEC_EXT.1.11.

For the length of the text-based pre-shared keys, a common length (22 characters) is required to help promote interoperability. If other lengths are supported they should be listed in the assignment; this assignment can also specify a range of values (e.g., "lengths from 5 to 55 characters") as well.

For FIA_PSK_EXT.1.3, the ST author fills in the method by which the text string entered by the administrator is "conditioned" into the bit string used as the key. This can be done by using one of the specified hash functions, or some other method through the assignment statement. If "bit-based pre-shared keys" is selected, the ST author specifies whether the TSF merely accepts bit-based preshared keys, or is capable of generating them. If it generates them, the requirement specified that they must be generated using the RBG specified by the requirements. If the TOE does not use bit-based pre-shared keys, the second

selection should be completed with "perform no other conditioning," as textbased pre-shared keys would then be the only type used.

Evaluation Activities ▼

> *FIA_PSK_EXT.1*
> **TSS**
> *The evaluator shall also examine the TSS to ensure it describes the process by which the bit-based preshared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1.*
>
> *The evaluator shall examine the TSS to ensure that it states that text-based pre-shared keys of 22 characters are supported. The evaluator shall also confirm that the TSS states the conditioning that takes place to transform the text-based pre-shared key from the key sequence entered by the user (e.g., ASCII representation) to the bit string used by IPsec, and that this conditioning is consistent with the FIA_PSK_EXT.1.3.*
>
> **Guidance**
> *If the TOE supports bit-based pre-shared keys, the evaluator shall confirm the operational guidance contains instructions for either entering bit-based pre- shared keys for each protocol identified in the requirement, or generating a bit-based pre-shared key (or both). The evaluator shall also examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1.*
>
> *The evaluator shall check that any management functions related to pre-shared keys that are performed by the TOE are specified in the operational guidance.*
>
> *The evaluator shall examine the operational guidance to determine that it provides guidance on the composition of strong text-based pre-shared keys, and (if the selection indicates keys of various lengths can be entered) that it provides information on the merits of shorter or longer pre-shared keys. The guidance must specify the allowable characters for pre-shared keys, and that list must include, at minimum, the same items contained in FIA_PSK_EXT.1.2.*
>
> **Tests**
> *The evaluator shall perform the following tests:*
> - **Test 1:** *The evaluator shall compose a pre-shared key of 22 characters that contains a combination of the allowed characters in accordance with the operational guidance, and demonstrates that a successful protocol negotiation can be performed with the key.*
> - **Test 2:** *[conditional]: If the TOE supports pre-shared keys of multiple lengths, the evaluator shall repeat Test 1 using the minimum length; the maximum length; and invalid lengths that are below the minimum length, above the maximum length, null length, empty length, or zero length. The minimum and maximum length tests should be successful, and the invalid lengths must be rejected by the TOE.*
> - **Test 3:** *[conditional]: If the TOE supports but does not generate bit-based pre-shared keys, the evaluator shall obtain a bit-based pre-shared key of the appropriate length and enter it per the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key.*
> - **Test 4:** *[conditional]: If the TOE does generate bit-based pre-shared keys, the evaluator shall generate a bit-based pre-shared key of the appropriate length and use it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key.*

Appendix C - Extended Component Definitions

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

C.1 Extended Components Table

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

Table 6: Extended Component Definitions

| Functional Class	Functional Components
Cryptographic Support (FCS)	FCS_IPSEC_EXT IPsec
Protection of the TSF (FPT)	FPT_TST_EXT TSF Self-Test
Identification and Authentication (FIA)	FIA_PSK_EXT Pre-Shared Key Composition
User Data Protection (FDP)	FDP_IFC_EXT Subset Information Flow Control

C.2 Extended Component Definitions

C.2.1 FCS_IPSEC_EXT IPsec

Family Behavior

Components in this family describe requirements for IPsec implementation.

Component Leveling

```
FCS_IPSEC_EXT  ├──────────┤1
```
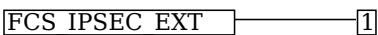
FCS_IPSEC_EXT.1, IPsec, requires the TSF to securely implement the IPsec protocol.

Management: FCS_IPSEC_EXT.1

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

- Specify VPN gateways to use for connections
- Specify IPsec VPN Clients to use for connections
- Specify IPsec-capable network devices to use for connections
- Specify client credentials to be used for connections

Audit: FCS_IPSEC_EXT.1

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

- Decisions to DISCARD or BYPASS network packets processed by the TOE
- Failure to establish an IPsec SA
- Establishment/Termination of an IPsec SA

FCS_IPSEC_EXT.1 IPsec

Hierarchical to: No other components.

Dependencies to: FCS_CKM.1 Cryptographic Key Generation

FCS_CKM.2 Cryptographic Key Distribution

FCS_COP.1 Cryptographic Operation

FCS_IPSEC_EXT.1.1

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

FCS_IPSEC_EXT.1.2

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

FCS_IPSEC_EXT.1.3

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

FCS_IPSEC_EXT.1.4

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

FCS_IPSEC_EXT.1.5

The The TSF shall implement the protocol: [**selection**:
- *IKEv1, using Main Mode for Phase I exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [***selection***: no other RFCs for extended sequence numbers, RFC 4304 for extended sequence numbers], [***selection***: no other RFCs for hash functions, RFC 4868 for hash functions], and [***selection***: support for XAUTH, no support for XAUTH],*
- *IKEv2 as defined in RFC 7296 (with mandatory support for NAT traversal as specified in section 2.23), RFC 8784, RFC 8247, and [***selection***: no other RFCs for hash functions, RFC 4868 for hash functions]*

].

FCS_IPSEC_EXT.1.6

The TSF shall ensure the encrypted payload in the [**selection**: *IKEv1*, *IKEv2*] protocol uses the cryptographic algorithms [*AES-CBC-128, AES-CBC-256 as specified in RFC 6379 and* [**selection**: *AES-GCM-128 as specified in RFC 5282*, *AES-GCM-256 as specified in RFC 5282*, *no other algorithm*].

FCS_IPSEC_EXT.1.7

The TSF shall ensure that [**selection**:
- *IKEv2 SA lifetimes can be configured by [***selection***: an Administrator, a VPN Gateway] based on [***selection***: number of packets/number of bytes, length of time] ,*
- *IKEv1 SA lifetimes can be configured by [***selection***: an Administrator, a VPN Gateway] based on [***selection***: number of packets/number of bytes, length of time] ,*
- *IKEv1 SA lifetimes are fixed based on [***selection***: number of packets/number of bytes, length of time]*

]. If length of time is used, it must include at least one option that is 24 hours or less for Phase 1 SAs and 8 hours or less for Phase 2 SAs.

FCS_IPSEC_EXT.1.8

The TSF shall ensure that all IKE protocols implement DH Groups [*19 (256-bit Random ECP), 20 (384-bit Random ECP), and* [**selection**: *24 (2048-bit MODP with 256-bit POS), 15 (3072-bit MODP), 14 (2048-bit MODP), no other DH groups*]].

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) number(s) of bits that is at least twice the "bits of security" value associated with the negotiated Diffie-Hellman group as listed in Table 2 of NIST SP 800-57, Recommendation for Key Management – Part 1: General*] bits.

FCS_IPSEC_EXT.1.10

The TSF shall generate nonces used in IKE exchanges in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^[**assignment**: *(one or more) "bits of security" value(s) associated with the negotiated Diffie-Hellman group as listed in Table 2 of NIST SP 800-57, Recommendation for Key Management – Part 1: General*].

FCS_IPSEC_EXT.1.11

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

FCS_IPSEC_EXT.1.12

The TSF shall not establish an SA if the [[**selection**: *IP address, Fully Qualified Domain Name (FQDN), user FQDN, Distinguished Name (DN)*] and [**selection**: *no other reference identifier type, [***assignment***: other supported reference identifier types]*]] contained in a certificate does not match the expected value(s) for the entity attempting to establish a connection.

FCS_IPSEC_EXT.1.13

The TSF shall not establish an SA if the presented identifier does not match the configured reference identifier of the peer.

FCS_IPSEC_EXT.1.14

The [**selection**: *TSF*, *VPN Gateway*] 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.

C.2.2 FPT_TST_EXT TSF Self-Test

Family Behavior

Components in this family describe requirements for self-test to verify functionality and integrity of the TOE.

Component Leveling
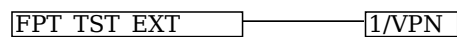
FPT_TST_EXT		1/VPN

FPT_TST_EXT.1/VPN, TSF Self-Test, requires the TOE to perform power on self-tests to verify its functionality and the integrity of its stored executable code.

Management: FPT_TST_EXT.1/VPN

No specific management functions are identified.

Audit: FPT_TST_EXT.1/VPN

There are no auditable events foreseen.

FPT_TST_EXT.1/VPN TSF Self-Test

Hierarchical to: No other components.

Dependencies to:

FPT_TST_EXT.1.1/VPN

The [**selection**: *TOE*, *TOE platform*] shall run a suite of self tests during initial start-up (on power on) to demonstrate the correct operation of the TSF.

FPT_TST_EXT.1.2/VPN

The [**selection**: *TOE*, *TOE platform*] shall provide the capability to verify the integrity of stored TSF executable code when it is loaded for execution through the use of the [**assignment**: *cryptographic services provided either by the portion of the TOE described by the Base-PP or by the operational environment*].

C.2.3 FIA_PSK_EXT Pre-Shared Key Composition

Family Behavior

Components in this family describes the requirements for pre-shared keys when implementing IPsec

Component Leveling

FIA_PSK_EXT		1

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

Management: FIA_PSK_EXT.1

No specific management functions are identified.

Audit: FIA_PSK_EXT.1

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.1 Pre-Shared Key Composition

Hierarchical to: No other components.

Dependencies to: FCS_IPSEC_EXT.1 IPsec

FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for IPsec.

FIA_PSK_EXT.1.2

The TSF shall be able to accept text-based pre-shared keys that:
- Are 22 characters and [**selection**: *[**assignment**: other supported lengths], no other lengths*],
- Composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", ")", and [**selection**: *no other special characters*, *[**assignment**: list of additional supported special characters]*].

FIA_PSK_EXT.1.3

The TSF shall condition the text-based pre-shared keys by using [**selection**: *SHA-1, SHA-256, SHA-512*, *[**assignment**: method of conditioning text string]*], [**selection**:
- *be able to [**selection**: accept, generate using the random bit generator specified in FCS_RBG_EXT.1],*
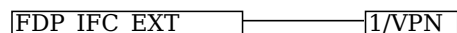- *perform no other conditioning*

].

C.2.4 FDP_IFC_EXT Subset Information Flow Control

Family Behavior

Components in this family describe the requirements that pertain to IP traffic and information flow through the VPN client.

Component Leveling

FDP_IFC_EXT		1/VPN

FDP_IFC_EXT.1/VPN, Subset Information Flow Control, requires the TSF to process all IP traffic through its VPN client functionality.

Management: FDP_IFC_EXT.1/VPN

No specific management functions are identified.

Audit: FDP_IFC_EXT.1/VPN

There are no auditable events foreseen.

FDP_IFC_EXT.1/VPN Subset Information Flow Control

Hierarchical to: No other components.

Dependencies to: FCS_IPSEC_EXT.1 IPsec

FDP_IFC_EXT.1.1/VPN

The TSF shall ensure that all IP traffic (other than IP traffic required to establish the VPN connection) flow through the IPsec VPN client.

Appendix D - Implicitly Satisfied Requirements

This appendix lists requirements that should be considered satisfied by products successfully evaluated against this PP. 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.

. **Table 7: Implicitly Satisfied Requirements**

Requirement	Rationale for Satisfaction
FCS_CKM.2 – Cryptographic Key Distribution, or FCS_COP.1 – Cryptographic Operation	FCS_CKM.1 (which is defined in this PP-Module as FCS_CKM.1/VPN) requires one of FCS_CKM.2 or FCS_COP.1 to be claimed so that the generated keys can serve some security-relevant purpose. Each of the Base-PPs for this PPModule define an iteration of FCS_COP.1 for symmetric cryptography that is expected to use the IKE keys generated by FCS_CKM.1/VPN. Therefore, this dependency is satisfied through requirements defined in the Base-PPs.
FCS_CKM.4 – Cryptographic Key Destruction	FCS_CKM.1 (which is defined in this PP-Module as FCS_CKM.1/VPN) requires FCS_CKM.4 to be claimed so that the generated keys are not disclosed through improper or nonexistent key destruction methods.
	Each of the supported Base-PPs except for the App PP define FCS_CKM_EXT.4 as an extended SFR, which defines key destruction functionality consistent with FCS_CKM.4, but with additional details that are specific to the respective technology types of the Base-PP. When the App PP is the Base-PP, this PP-Module defines its own instance of FCS_CKM_EXT.4 to achieve the same purpose. The dependency on FCS_CKM.4 is considered to be satisfied through the fact that a compliant TOE will always claim FCS_CKM_EXT.4, which is intended to satisfy the same purpose.
FCS_COP.1 – Cryptographic Operation	FCS_IPSEC_EXT.1 has a dependency on FCS_COP.1 because of the cryptographic operations that are needed in support of implementing the IPsec protocol. FCS_COP.1 is not defined in this PP-Module because each of the supported Base-PPs define iterations of FCS_COP.1 that support the functions that are relevant to IPsec.
FMT_MTD.1 – Management of TSF Data	FAU_SEL.1/VPN has a dependency on FMT_MTD.1 to enforce appropriate access controls on the audit configuration, as this is TSF data. This SFR is not explicitly defined in any of the supported Base-PPs but the dependency is implicitly addressed by each Base-PP in the following manner: • GPOS PP: The GPOS PP implicitly defines the existence of 'user' and 'administrator' roles in the extended SFRs FMT_MOF_EXT.1 and FMT_SMF_EXT.1. A TOE that conforms to this BasePP can associate the ability to perform the functionality defined by FAU_SEL.1/VPN to one or both of these roles. • MDF PP: The GPOS PP implicitly defines the existence of 'user,' 'administrator,' and 'MDM' roles in the extended SFRs FMT_MOF_EXT.1 and FMT_SMF_EXT.1. A TOE that conforms to this BasePP can associate the ability to perform the functionality defined by FAU_SEL.1/VPN to one or more of these roles. • App PP: The App PP does not define the existence of a separately authenticated management interface; instead, the App PP assumes that authentication to the underlying OS platform is sufficient authorization to access the application's management functionality. • MDM PP: The MDM PP defines the existence of management roles in FMT_SMR.1(1). A TOE that conforms to this Base-PP can associate the ability to perform the functionality defined by FAU_SEL.1/VPN to one or more of the roles defined here.
FPT_STM.1 – Reliable Time Stamps	FAU_GEN.1/VPN has a dependency on FPT_STM.1 because audit records are required to have timestamps that are based on reliable clock data. All of the supported Base-PPs either define this requirement explicitly or provide rationale for why the reader to expect that a reliable clock service is expected to be present. Depending on the claimed Base-PP, the dependency is satisfied in the following manner: • GPOS PP: The GPOS PP states that FPT_STM.1 is implicitly satisfied by the requirements of FAU_GEN.1 since that requirement could not be satisfied if no clock service was present. Additionally, a clock service is reasonably assumed to be provided by a general-purpose OS. • MDF PP: The MDF PP explicitly defines FPT_STM.1. • App PP: The App PP assumption A.PLATFORM assumes that the general-purpose

computing platform on which the TOE is installed is 'a trustworthy computing platform.' System time data is not explicitly mentioned but a clock service is reasonably assumed to be provided by a generalpurpose computer.
- MDM PP: The MDM PP assumption A.MDM_SERVER_PLATFORM assumes that the platform on which the TOE is installed will provide reliable time services.

FPT_STM.1 – Reliable Time Stamps	FAU_GEN.1 has a dependency on FPT_STM.1. While not explicitly stated in the PP, it is assumed that this will be provided by the underlying hardware platform on which the TOE is installed. This is because the TOE is installed as a software or firmware product that runs on general-purpose computing hardware so a hardware clock is assumed to be available.
FPT_STM.1 – Reliable Time Stamps	FIA_X509_EXT.1 has a dependency on FPT_STM.1. While not explicitly stated in the PP, it is assumed that this will be provided by the underlying hardware platform on which the TOE is installed. This is because the TOE is installed as a software or firmware product that runs on general-purpose computing hardware so a hardware clock is assumed to be available.

Appendix E - Entropy Documentation and Assessment

The TOE does not require any additional supplementary information to describe its entropy source(s) beyond the requirements outlined in the Base-PPs. As with other Base-PP requirements, the only additional requirement is that the entropy documentation also applies to the specific VPN client 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
HTTP	HyperText Transfer Protocol
OE	Operational Environment
PP	Protection Profile
PP-Configuration	Protection Profile Configuration
PP-Module	Protection Profile Module
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SSL/TLS	Secure Sockets Layer/Transport Layer Security
ST	Security Target
STIP	SSL/TLS Inspection Proxy
TA	Trust Anchor (Trust Store)
TOE	Target of Evaluation
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification
URL	Uniform Resource Locator

Appendix G - Bibliography

Identifier	Title
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017. • Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017. • Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.
[App PP]	Protection Profile for Application Software, Version 1.3, March 2019
[MD PP]	Protection Profile for Mobile Device Fundamentals, Version 3.1, June 2017
[MDM PP]	Protection Profile for Mobile Device Management (This needs to be updated) , Version 3.1, June 2017
[OS PP]	Protection Profile for General Purpose Operating Systems, Version 4.2.1, April 2019
[SD]	Supporting Document Mandatory Technical Document, PP-Module for Virtual Private Network (VPN) Clients, Version 2.1, November 2019