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).

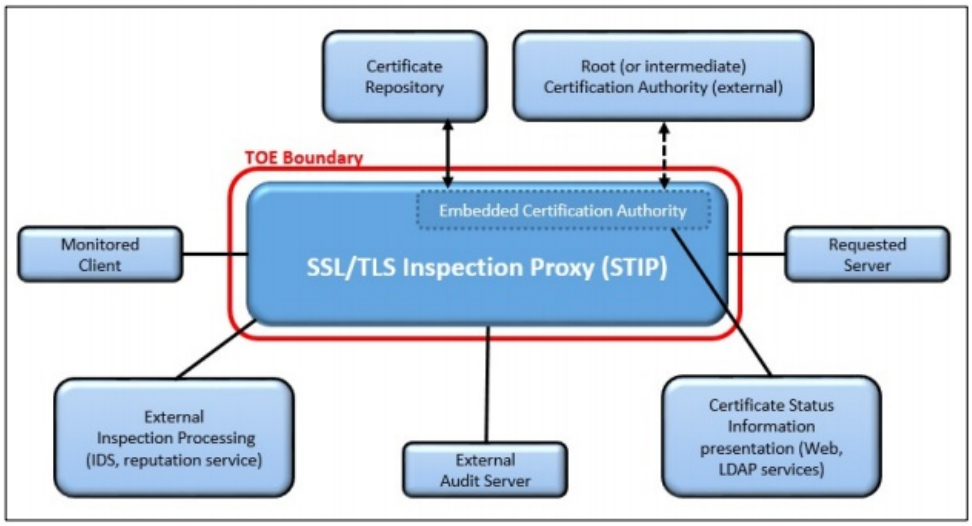


Figure 1: TLS Inspection Infrastructure

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.

Table 1: Security Objectives Rationale

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.

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.

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.

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.

5.1.1.3 Identification and Authentication (FIA)

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.

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.

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.

5.1.1.4 Trusted Path/Channels (FTP)

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*].

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

5.2.2 Security Audit (FAU)

FAU_GCR_EXT.1 Generation of Certificate Repository

FAU_GCR_EXT.1.1

The TSF shall [**selection**: *store*, *invoke the Operational Environment to store*] certificates issued by the TSF.

Application Note: While there is a requirement that a certificate repository exists and the TOE stores all certificates it generates in that repository, the repository can physically be within the TOE or in the OE. If the repository is provided by the TOE, then the first item in the first selection is chosen. If the storage is provided by the OE, then the second item in the first selection is chosen. It should be noted that the physical implementation of the certificate repository is left to the vendor; for instance, it can be a standalone store, or incorporated within the audit trail.

FAU_STG.4 Prevention of Audit Data Loss

FAU_STG.4.1

The TSF shall [*prevent audited events, except those taken by the **[assignment: Security Administrator, Auditor]***] and [**assignment**: *other actions to be taken in case of audit storage failure*] if the audit trail **cannot be written to**.

Application Note: This requirement applies to the TOE regardless of whether the audit trail is stored within the TOE boundary or on an external system in the Operational Environment. If the audit trail is stored locally, then the requirement applies when the audit trail cannot be written to when it is full. If the audit trail (in whole or in part) is stored on a system external to the TOE, then the requirement applies when the connection between the TOE and the external audit server becomes disconnected and the audit trail cannot be written to. In the case where the audit trail is external to the TOE and cannot be written to because it is full (and the TOE has some way of detecting that), then the requirement applies in that case as well. In all cases, the ST author is expected to describe (in the TSS) how the TSF is made aware of any such failures and how it behaves in response.

5.2.3 Cryptographic Support (FCS)

FCS_COP.1/STIP Cryptographic Operation (Data Encryption/Decryption in Support of STIP

FCS_COP.1.1/STIP

The TSF shall perform encryption/decryption in accordance with a specified cryptographic algorithms[*AES in CCM and CCM-8 mode and [**selection**: TDES used in CBC mode with 3 distinct keys in its key set, no other algorithms]*] and cryptographic key sizes [**selection**: *128 bits*, *192 bits*, *256 bits*] that meet the following: [*AES as specified in ISO 18033-3, CCM and CCM-8 as specified in NIST SP 800-38C and [**selection**: TDES as specified in NIST SP 800-67 Rev 2 and CBC mode as specified in NIST SP 800-38A addendum, no other standards]*].

Application Note: This requirement, in conjunction with FCS_COP.1/DataEncryption from the BasePP, is used to support FCS_TTTC_EXT.1 and FCS_TTTS_EXT.1. Note that FCS_TTTC_EXT.1 and FCS_TTTS_EXT.1 may necessitate certain selections.

FCS_STG_EXT.1 Cryptographic Key Storage

FCS_STG_EXT.1.1

Persistent private and secret keys shall be stored within the TSF using [**assignment**: *method of hardware-protected storage*].

Application Note: This requirement ensures that persistent secret keys and private keys are stored securely when not in use. Methods of hardware protected storage can be direct or via encryption with a KEK which is protected by hardware. The application notes for FPT_KST_EXT.2.1 contain further discussion of private and secret keys referenced by this SFR

FCS_TTTC_EXT.1 Thru-Traffic TLS Inspection Client Protocol

FCS_TTTC_EXT.1.1

The TSF shall implement [*TLS 1.2 (RFC 5246), TLS 1.0 (RFC 2246), and [**selection**: TLS 1.1 (RFC 4346), no other TLS versions]*] as a client to the requested server that supports the following cipher suites: [

- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*

- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_DHE_RSA_ WITH_AES_256_CCM as defined in RFC 6655*
- *TLS_RSA_ WITH_AES_256_CCM as defined in RFC 6655*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_DHE_RSA_WITH_AES_128_CCM as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_RSA_WITH_AES_128_CCM as defined in RFC 6655*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246*
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5426*
- *TLS_RSA_WITH_AES_128_CCM_8 as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_128_CCM_8 as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_256_CCM_8 as defined in RFC 6655*
- *TLS_RSA_WITH_AES_256_CCM_8 as defined in RFC 6655*
- *[**selection**: TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 8422, TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 5246, TLS_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 5246, [**assignment**: other supported cipher suites], no other cipher suites]*

] and also supports functionality for [**selection**:

- *mutual authentication*,
- *session renegotiation*,
- *neither mutual authentication nor session renegotiation*

].

Application Note: TLS version 1.2 and 1.0 must be supported; support for TLS version 1.1 is optional, and should be chosen if the STIP supports it. The list of cipher suites to support is mandatory but includes some selections in order to support legacy servers that may be required by the monitored clients; additional cipher suites can be included in the assignment. The order of the cipher suites above should be maintained in the ST; FCS_TTTC_EXT.1.4 indicates that the cipher suites are presented in order of preference in the Client Hello sent to the requested server, and that preference is defined as the order in the above SFR.

The above list (as instantiated in the ST) limits the cipher suites that may be proposed by the TOE to the requested server. Behavior if the requested server responds with a cipher suite that is not in the list is defined in FDP_TEP_EXT.1.8.

The selection should indicate if mutual authentication and/or session renegotiation is supported. These selections must be the same for both FCS_TTTC_EXT.1.1 and FCS_TTTS_EXT.1.1. If mutual authentication is selected, the requirements in Section B.4 will be included by the ST author. For this technology, mutual authentication is not desirable on these connections because the STIP will have to issue a certificate representing the client to the requested server, and the server will have to have a trust anchor for that certificate. If session renegotiation is selection, FCS_TTTC_EXT.4 from Section B.5 will be included by the ST author.

The data encryption and decryption algorithms used in this element are performed in accordance with FCS_COP.1/STIP.

FCS_TTTC_EXT.1.2

The TSF shall verify that the presented identifier matches the reference identifier of the server requested by the monitored client using methods described in RFC 6125 section 6 for DNS name types, and via exact, byte-by-byte

matching for IP address name types.

Application Note: The rules for verification of identity are described in Section 6 of RFC 6125. The monitored client may specify the server name in the SNI extension of the Client Hello, or via some other method (e.g., DNS lookup) supported by the TSF. The method for determining that the identity presented matches that expected by the client should be fully described in the ST.

Additionally, support for use of IP addresses in the Subject Name or Subject Alternative Name of TLS server certificates is discouraged as against best practices but may be implemented by requested servers. When no DNS name type reference ID is available from the monitored client and the certificate presented by the requested server includes an IP address name type, exact byte-by-byte matching of the IP address to an IP address reference ID is required. If the certificate does not contain an identifier of type IP address, and no other name type is included as a reference Identifier, the IP address from the underlying transport layer protocol between the TSF and the requested servers should match the IP address reference identifier.

FCS_TTTC_EXT.1.3

The TSF shall validate the certificate presented by the server and terminate the connection if the certificate is invalid, except as allowed by FIA_X509_EXT.2.2.

Application Note: Validity is determined by the identifier verification, certificate path, the expiration date, processing of critical extensions, and the revocation status in accordance with RFC 5280. Certificate validity is specified by and tested in accordance with FIA_X509_EXT.1/STIP. The result of the checks will be one of 1) the certificate is valid; 2) the certificate is invalid; 3) the validity of the certificate is indeterminate because a connection cannot be established to check the revocation status of the certificate (but all other validity checks have passed). FCS_X509_EXT.2.2 (in conjunction with FDP_TEP_EXT.1.5) indicates what the TSF is supposed to do if a connection cannot be established to check the revocation status for this connection (the TOE to a requested server).

FCS_TTTC_EXT.1.4

The TSF shall formulate the Client Hello such that it presents the highest version of the TLS protocol supported by the proxy function in the version field, and presents the list of cipher suites in descending order of preference associated with requested server.

Application Note: This applies to the initial Client Hello sent to the requested server. This may result in a connection being established for an inspect operation, or may not lead to a connection if a bypass or block operation is determined. It should be noted that this transaction may be made even though the result will eventually be block or bypass, because the rule (see FDP_TEP_EXT.1) may require the verified identity of the server, so this connection would be required so that the server certificate could be obtained and verified.

FCS_TTTC_EXT.5 Thru-Traffic TLS Inspection Client Support for Supported Groups Extension

FCS_TTTC_EXT.5.1

The TSF shall present the Supported Groups Extension in the Client Hellow with the supported groups **[selection:**

- ***secp256r1*,**
- ***secp384r1*,**
- ***secp521r1*,**
- ***ffdhe2048(256)*,**
- ***ffdhe3072(257)*,**
- ***ffdhe4096(258)*,**
- ***ffdhe6144(259)*,**
- ***ffdhe8192(260)*,**
- ***[assignment: other supported curves]]***

] .

Application Note: Since support for all of the cipher suites listed in FCS_TTTC_EXT.1.1 is required, at least one of the curves and one of the finite field groups must be chosen by the ST author as appropriate for the cipher suites and the implementation.

If additional elliptic curves are supported, ST author should describe the elliptic curve parameters for each supported elliptic curve in the assignment in accordance with RFC 7919. No additional Diffie-Hellman groups should be claimed in the assignment.

The Supported Groups Extension was previously referred to as the Supported Elliptic Curves Extension and is described in RFC 7919.

Since a requested server session might not adhere to RFC 7919 processing rules, the TOE should accept additional DH groups that might be presented in the requested server's key exchange message.

FCS_TTTS_EXT.1 Thru-Traffic TLS Inspection Server Protocol

FCS_TTTS_EXT.1.1

The TSF shall implement [*TLS 1.2 (RFC 5246), TLS 1.0 (RFC 2246), and [**selection**: TLS 1.1 (RFC 4346), no other TLS versions]*] as a server to the monitored client that supports the following cipher suites: [

- *TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_RSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_256_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289*
- *TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5289*
- *TLS_DHE_RSA_ WITH_AES_256_CCM as defined in RFC 6655*
- *TLS_RSA_ WITH_AES_256_CCM as defined in RFC 6655*
- *TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_DHE_RSA_WITH_AES_128_CCM as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5289*
- *TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289*
- *TLS_RSA_WITH_AES_128_CCM as defined in RFC 6655*
- *TLS_RSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5288*
- *TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_DHE_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246*
- *TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA as defined in RFC 8422*
- *TLS_RSA_WITH_AES_256_CBC_SHA as defined in RFC 5246*
- *TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA as defined in RFC 8422*
- *TLS_DHE_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5246*
- *TLS_RSA_WITH_AES_128_CBC_SHA as defined in RFC 5426*
- *TLS_RSA_WITH_AES_128_CCM_8 as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_128_CCM_8 as defined in RFC 6655*
- *TLS_DHE_RSA_WITH_AES_256_CCM_8 as defined in RFC 6655*
- *TLS_RSA_WITH_AES_256_CCM_8 as defined in RFC 6655*
- *[**selection**: TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 8422, TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 5246, TLS_RSA_WITH_3DES_EDE_CBC_SHA as defined in RFC 5246, [**assignment**: other supported cipher suites], no other cipher suites]*

] and also supports functionality for [**selection**:

- *mutual authentication*,
- *session renegotiation*,
- *neither mutual authentication nor session renegotiation*

].

Application Note: TLS version 1.2 and 1.0 must be supported; support for TLS version 1.1 is optional, and should be chosen if the STIP supports it. The list of cipher suites to support is mandatory but includes some selections in order to support legacy clients that may be required by the organization; additional cipher suites can be included in the assignment.

The above list (as instantiated in the ST) limits the cipher suites that may be specified by the TOE when responding to the monitored client. The data encryption and decryption algorithms used in this element are performed in accordance with FCS_COP.1/STIP.

The selection should indicate if mutual authentication and/or session renegotiation is supported. These selections must be the same for both FCS_TTTC_EXT.1.1 and FCS_TTTS_EXT.1.1. If mutual authentication is selected, the requirements in Section B.4 will be included by the ST authors. For this technology, mutual authentication is not desirable on these connections because the STIP will have to issue a certificate representing the client to the requested server, and the server will have to have a trust anchor for that certificate. If session renegotiation is selection, FCS_TTTS_EXT.4 in section B.5 will be included by the ST authors.

The data encryption and decryption algorithms used in this element are performed in accordance with FCS_COP.1/STIP.

FCS_TTTS_EXT.1.2

The TSF shall deny connections from clients requesting [SSL 2.0, SSL 3.0, and [**selection**: *TLS 1.1, no other SSL or TLS versions*] for through-traffic processing.

Application Note: All SSL versions are denied regardless of exception specifications. Any TLS versions not selected in FCS_TTTS_EXT.1.1 should be selected here

FCS_TTTS_EXT.1.3

The TSF shall perform key establishment for TLS with a monitored client using [

- *RSA with key size 2048 bits, [**selection**: 1024 bits, 1536 bits, 3072 bits, 4096 bits, no other sizes]*
- *[**selection**:*
  - *Diffie-Hellman parameters of size 2048 bits, [**selection**: 1024 bits, 1536 bits, 3072 bits, 4096 bits, 8192 bits, no other sizes] ,*
  - *Diffie-Hellman groups ffdhe2048, [**selection**: ffdhe3072, ffdhe4096, ffdhe6144, ffdhe8192, no other groups]*

  ]
- *EC Diffie-Hellman parameters using elliptic curves [**selection**: secp256r1, secp384r1, secp521r1, [**assignment**: other curves]] and no other curves*

.

Application Note: The selections in this element should indicate all key establishment sizes and/or groups supported.

5.2.4 User Data Protection (FDP)

FDP_CER_EXT.1 Certificate Profiles for Server Certificates

FDP_CER_EXT.1.1

The TSF shall implement a certificate profile function for TLS server certificates issued by a CA embedded within the TOE, and shall ensure that issued certificates are consistent with configured profiles.

FDP_CER_EXT.1.2

The TSF shall generate certificates using profiles that comply with requirements for certificates as specified in IETF RFC 5280, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile” as refined below. At a minimum, the TSF shall ensure that:

a. The version field shall contain the integer 2.
b. The issuerUniqueID or subjectUniqueID fields are not populated.
c. The serialNumber shall be unique with respect to the issuing Certification Authority.
d. The validity field shall specify a notBefore value that does not precede the current time and a notAfter value that does not precede the value specified in notBefore.
e. The issuer field is not empty and is populated with the **[selection: *Security Administrator, CA Operations Staff*]**-configured CA name.
f. The signature field and the algorithm in the subjectPublicKeyInfo field shall contain the OID for a signature algorithm specified in FCS_COP.1/SigGen in the NDcPP.
g. The following extensions are supported:
   a. authorityKeyIdentifier
   b. keyUsage
   c. extendedKeyUsage
   d. certificatePolicy
   e. [**selection**: *subjectKeyIdentifier*, *basicConstraints*, *cRLDistributionPoints*, *authorityInfoAccess*, *no other extensions*]
h. A subject field containing a null Name (e.g., a sequence of zero relative

distinguished names) is accompanied by a populated critical subjectAltName extension.

i. The authorityKeyIdentifier extension in any certificate issued by the TOE must be populated and must be the same as the subjectKeyIdentifier extension contained in the TOE's embedded CA's signing certificate

j. Populated keyUsage and extendedKeyUsage fields in the same certificate shall contain consistent values reflecting exclusive TLS server use as follows:

keyUsage	extendedKeyUsage
digitalSignature	serverAuth
digitalSignature, keyEncipherment	serverAuth
digitalSignature,keyAgreement	serverAuth

k. [**selection**: *The subjectKeyIdentifier extension is populated with a value unique for each public key contained in a certificate issued by the TSF, no other constraints*]

Application Note: RFC updates to RFC 5280 are included in this requirement. The inclusion of the cRLDistributionPoints and authorityInfoAccess extensions depend on the selections made in FDP_CSI_EXT.1.3.

Uniqueness for the subject key identifier (item k above) is specific to the instance of the embedded CA. The same configured CA should not issue certificates with different public keys having the same subject key identifier.

If subjectKeyIdentifier is chosen in the selection in item g, then the ST author selects the first selection in item k; otherwise, select "no other constraints."

FDP_CER_EXT.1.3

The TSF shall implement the following rules for populating certificate fields based on constraints imposed by the TOE's embedded CA's signing certificate:

- The validity field shall specify a notAfter time that does not exceed the notAfter time of the CA's signing certificate.
- The issuer field identifies the [**selection**:
  - *subject,*
  - *[**assignment**: **[selection: Security Administrator, CA Operations Staff]**-assigned identifying information]*

  ] of the CA's signing certificate.
- [**selection**:
  - *The subject name is limited by name constraints specified in the CA's signing certificate,*
  - *[**assignment**: list of rules],*
  - *no other rules*

  ]

FDP_CER_EXT.1.4

The TSF shall implement the following rules for populating certificate fields based on the validated certificate and constraints imposed by the **[selection: *Security Administrator, CA Operations Staff*]**:

a. The Subject/Subject Alternative Name shall be copied from validated server certificate.

b. The notBefore field shall not precede the notBefore field of the validated server certificate.

c. The notAfter field shall not exceed the notAfter field of the validated server certificate.

d. The notAfter field shall not exceed the current time by more than a maximum validity duration value as configured by a **[selection: *Security Administrator, CA Operations Staff*]** user.

e. If the basicConstraints field is configured to be present, it shall be populated with the value cA=False

f. The subject public key shall be generated in accordance with FCS_CKM.1.1 **in the NDcPP.**

g. [**selection**:
  - *policy OID/policy mapping fields are populated in accordance with [**assignment**: a **[selection: Security Administrator, CA Operations Staff]** configured mapping from validated server certificate values to one or more stated policy OIDs],*
  - *[**assignment**: list of rules],*
  - *no additional rules*

]

Application Note: It is preferred that a new public key be generated each time a certificate is generated.

FDP_CER_EXT.2 Certificate Request Matching of Server Certificates

FDP_CER_EXT.2.1

The TSF shall establish and record a linkage from validated certificates to issued certificates.

Application Note: This requirement ensures that the TOE provides linkage between TLS server certificates validated during a TLS session establishment by the TOE and resulting certificates issued by the TOE to represent the requested server (or monitored client if supported). In terms of Certification authority operations, an automatically approved certificate request is implied by the validated certificate and the configured TLS session establishment policy identified in FDP_TEP_EXT.1.

FDP_CER_EXT.3 Certificate Issuance Rules for Server Certificates

FDP_CER_EXT.3.1

The TSF shall issue certificates in response to a validated server certificate according to the following rules: The issued certificate is in compliance with a current certificate profile defined in accordance with FDP_CER_EXT.1 and

- The TLS session establishment policy is configured to allow inspection of TLS sessions between monitored clients and a requested server authenticated to the TSF by the validated certificate,

[**selection**:

- *A valid certificate for the same subject is not present in cache*,
- *The embedded CA certificate's name space allows issuance of a certificate that represents the authenticated server*

].

FDP_CER_EXT.3.2

The TSF shall reject all certificate requests originating external to the TOE.

FDP_CSIR_EXT.1 Certificate Status Information Required

FDP_CSIR_EXT.1.1

The TSF shall [**selection**: *generate certificate status information*, *only issue certificates with validity period of less than [24 hours]*].

Application Note: Based on the selection, the ST author must choose the appropriate requirements from Appendix B.1 of this PP-Module.

The ST should specify whether certificate status information is generated. If the TSF can be configured so the validity of issued certificates is longer than 24 hours, certificate status information must be able to be generated.

Certificate policies associated with the issuance of TLS server certificates imply that certificates issued by the TSF must be revoked within a certain time period of discovering they do not properly represent the asserted subject. Certificate status information is not required if the validity period of any issued certificate is less than the time in which this status information must be provided. Even for emergency revocations, this time period is typically greater than 24 hours.

FDP_PPP_EXT.1 Plaintext Processing Policy

FDP_PPP_EXT.1.1

The TSF shall enforce the TLS plaintext processing policy on information flows containing plaintext produced by inspection processing of the TOE between TLS session termination points and [**selection**: *distinct internal inspection processing functional components*, *internal inspection processing functional components and an interface to external inspection processing environment*]

Application Note: This element identifies the policy (TLS plaintext processing) that is applied to decrypted TLS session data received by the TSF via an external interface for which the TLS session establishment policy, FDP_TEP_EXT.1, determines inspection processing is authorized, resulting in the exposure of the underlying plaintext associated to the TLS session. Information flows containing such data are referred to as TLS session threads.

Every network packet decrypted under the TLS session establishment policy inspection operation is associated to a TLS session thread and has the ruleset

that expresses this policy applied between each distinct inspection processing functional component, including the points where TLS encryption/decryption occurs. This PP-Module allows both internal and external inspection processing functional components. Internal inspection processing components, if supported, range from simple routing functions that determine whether to abort inspection processing of a TLS session thread based on an identifier, to complicated intrusion detection/prevention functions. External inspection processing components, if supported, are accessed via a controlled interface of the TOE to a protected computing environment, considered as part of the operational environment.

FDP_PPP_EXT.1.2

The TSF shall allow the definition of TLS plaintext processing policy rules using [**assignment**: *entity attributes of the requested server*], [**assignment**: *indicators of inspection processing results*] and distinct interfaces.

Application Note: This element identifies the various attributes that are applicable when constructing rules to be enforced by this requirement. The attributes to be included in this requirement include those which are exposed only for TLS sessions undergoing inspection processing in accordance with FDP_TEP_EXT.1 after the TLS application payload is decrypted, or can simply be the thread indicator to implicitly include attributes obtained by the TLS session establishment policy. Indicators can include specific error alerts from an internal inspection processing functional component, or can be a timeout resulting from an inspection processing functional component blocking traffic requiring no explicit signaling.

FDP_PPP_EXT.1.3

The TSF shall allow the following operations to be associated with the plaintext processing policy: permit, block, and [**selection**: *bypass*, *no other operation*], with the capability to log the operation.

Application Note: This element defines the operations that can be associated with rules used to manage inspection processing of TLS session threads. Permit allows the information flow to continue between inspection processing functional components; bypass indicates that the information flow is not processed by the processing component, but is forwarded to either the TLS encryption/decryption buffer, or the next plaintext inspection functional processing component; block drops subsequent information flows associated to the TLS session thread and informs the TLS session establishment policy to transition the TLS session to a Block Operation for subsequent TLS messages to or from the monitored client or requested server. It is permissible to use timeouts as indicators between inspection processing functional components or between the TLS plaintext processing policy and the TLS session establishment policy. Note this requirement does not specify the behavior of the inspection processing functional components, as this functionality is out of scope of this PP-Module. It only specifies the policy controlling the TOEs response to indicators from those processing components or to take advantage of the requested server’s subject attributes exposed by decryption

FDP_PPP_EXT.1.4

The TSF shall allow the Plaintext Processing Policy to be applied at each information flow control point between inspection processing functional components, including any network interface used to support external inspection processing.

Application Note: This element indicates where the TLS plaintext processing policy can be assigned. A conforming TOE must be able to assign processing rules to prevent TLS data from being exposed to unauthorized processing units based on the requested server attributes, and to allow TLS sessions containing malicious or unauthorized data, as determined by the inspection processing functional components, to be blocked at the earliest possible point, avoiding compromise of the TOE or the monitored client.

FDP_PPP_EXT.1.5

The TSF shall

- drop Information flows between inspection processing components, including any interface to external inspection processing components, that cannot be associated to an existing TLS session thread.
- inform the TLS session establishment policy of the TLS session thread associated to any information flow that is blocked by the plaintext processing policy.

Application Note: This element identifies state information shared by the TLS inspection processing policy and the TLS session establishment policy associated. The TSF may inform the TLS session establishment policy that it has

blocked a data flow either explicitly, by sharing state information, signaling, or other mechanism, or implicitly via the use of time-out mechanisms.

FDP_PRC_EXT.1 Plaintext Routing Control

FDP_PRC_EXT.1.1

The TSF shall control the routing of information flows containing plaintext within a TLS session thread in accordance with the configured Plaintext Processing Policy identified in FDP_PPP_EXT.1.

FDP_PRC_EXT.1.2

The TSF shall separate information flows containing plaintext within different TLS session threads.

FDP_PRC_EXT.1.3

The TSF shall not expose plaintext within a TLS session thread except to inspection processing functional components identified in, and as authorized by the configured Plaintext Processing Policy, as described in FDP_PPP_EXT.1.

FDP_RIP.1 Subset Residual Information Protection

FDP_RIP.1.1

The TSF shall ensure that any previous information content of a resource is made unavailable upon the [**selection**: *allocation of the resource to, deallocation of the resource from*] the following objects: [**assignment**: *list of objects*].

Application Note: “Resources” in the context of this requirement are any data buffers used to implement STIP functions, including the TLS buffers containing decrypted TLS payloads. The concern is that a buffer or memory area might be reused in subsequent function or communication channel resulting in inappropriate disclosure of sensitive data. “Objects” refers to any sensitive data objects that are under control of the TSF.

FDP_STG_EXT.1 Certificate Data Storage

FDP_STG_EXT.1.1

The TSF shall use [**selection**: *access controlled storage, an integrity mechanism*] to protect the trusted public keys and certificates (trust store elements) used to validate local logon, trusted channel, and external communication to the STIP.

Application Note: If “an integrity mechanism” is selected, FCS_CKM_EXT.5 must be included in the ST

FDP_STIP_EXT.1 SSL/TLS Inspection Proxy Functions

FDP_STIP_EXT.1.1

The TSF shall be capable of performing the Inspection Operation consisting of establishing a TLS session between TOE and the requested server according to FCS_TTTC_EXT.1, establishing a TLS session between the monitored client and the TOE according to FCS_TTTS_EXT.1, and routing decrypted application data from either of these TLS sessions to or between inspection processing functional components within the TOE, or between the TOE and external inspection processing functional components to a unique TLS session thread, according to FDP_PPP_EXT.1 and FDP_PRC_EXT.1.

Application Note: This defines the inspection operation, where the TLS connection is terminated at both ends on the TOE and the opportunity for inspection of the contents is allowed.

FDP_STIP_EXT.1.2

The TSF shall obtain a certificate from the TOE CA that represents the requested server for establishment of the TLS session with the monitored client when performing an inspect operation.

Application Note: Certificates are generated by the TOE’s embedded CA function, or obtained from an optional certificate cache maintained by the TOE. Certificate caching is not required, however, in the case where certificate caching is supported, the TSF will still need to perform certificate generation if no corresponding cache entry can be found for the requested server that matches the current certificate profile.

FDP_STIP_EXT.1.3

The TSF shall [**selection**: *require administrator confirmation of consent, provide a consent to monitor banner to the client, in accordance with FTA_TAB.1**/TLS**, and receive an affirmative response*] prior to sending decrypted TLS application data from a monitored client to inspection processing functional component as

part of an inspection operation.

Application Note: The selection “require administrator confirmation of consent” means that there is a means for the administrator to approve the operation based on receiving consent from the monitored client(s). This would include “real-time” approval mechanisms (a pop-up, for instance, accessible to an administrator) as well as configuration settings indicating “pre-approval” (again, only accessible by the administrator) such as one-time approval at installation (prior to any decryption), or included in a logon banner for administrators. A particular mechanism is not specified as it is up to the implementation. The intent is simply to ensure consent is obtained prior to monitoring.

FDP_STIP_EXT.1.4

The TSF shall provide the Bypass operation functionality by forwarding traffic between the monitored client and requested server such that monitored client can establish and maintain a TLS connection to the requested server.

Application Note: This merely defines the Bypass operation, where the STIP does not inspect the traffic, and just forwards packets between the monitored client interface and the requested server interface.

FDP_STIP_EXT.1.5

When initiating a Block operation, the TSF shall be capable of providing a [**selection**: *TLS error response*, *[**assignment**: other error message]*] to the monitored client associated with the blocked TLS session.

Application Note: This requires the TOE to provide some form of notification to monitored client when the monitored client attempts to initiate a connection and that connection is blocked. This can be done through the TLS error response, or (using the “assignment” part of the selection) some other means defined by the ST author.

FDP_TEP_EXT.1 SSL/TLS Inspection Proxy Policy

FDP_TEP_EXT.1.1

The TSF shall perform SSL/TLS Inspection Proxy functions and enforce SSL/TLS Inspection Proxy rules on TLS traffic received by the TSF from monitored clients and servers requested by monitored clients, and on TLS traffic controlled by the TSF to be sent to monitored clients and servers requested by monitored clients.

Application Note: This element defines the policy and requires the rules (defined in other elements of this component) to be applied to TLS network traffic from monitored clients to requested servers that is processed at the TOE’s network interfaces (as required in subsequent elements).

This requirement is to be enforced even if the network interfaces are saturated/overwhelmed with network traffic.

The requirement only applies to network traffic at the external interfaces that is identified as TLS traffic between a monitored client and requested server. This does not apply, for instance, to TLS traffic associated with administration of the STIP.

FDP_TEP_EXT.1.2

The TSF shall allow the definition of SSL/TLS Inspection Proxy rules based on the following attributes of each monitored client and requested server: [

- *Network Protocol fields: [**selection**: IPv4, IPv6, [**assignment**: other internet protocol]]:*
  - *Source address*
  - *Destination address*
  - *Source port*
  - *Destination port*
  - *[**selection**: [**assignment**: other fields containing identity attributes for the monitored client or requested server], no other fields]*
- *TLS Client Hello handshake message:*
  - *Server_name extension of the requested server*
  - *Client side interface*
- *TLS Server Certificate message:*
  - *Issuer*
  - *Subject*
  - *SubjectAlternateName*
- *Distinct interface:*
- *[**selection**:*

- *TLS client certificate message [**selection**:*
  - *Certificate issuer,*
  - *Certificate subject,*
  - *Certificate subject alt name*

  ] ,
- *[**assignment**: other attributes],*
- *no other attributes*

]

].

Application Note: This element identifies the various attributes that are applicable when constructing rules to be enforced by this requirement. The rules apply to external interfaces receiving TLS messages from a monitored client (client side interface), and to the TLS messages received by the TOE in response to the TSF initiating a TLS connection to a server requested by the monitored client (server side interface).

Network Protocol fields are used by the TSF to determine the IP address of the monitored client and the IP address of the requested server in traffic received on the client side interface only. Indicate which network protocols, including the internet layer and transport layer protocols that are used to determine the indicated fields that can be applicable when constructing rules for this policy.

The TLS Client Hello messages, and optional client certificate messages are received on the client side interface only. If the TSF supports client authentication, 'TLS client certificate message' should be selected (with the appropriate subselections supported by the TOE), and FCS_TTTS_EXT.3, Authentication of Monitored Clients should be claimed.

The TLS certificate message is received on a server side interface prior to the TSF sending a Server Hello done message on the client side interface.

FDP_TEP_EXT.1.3

The TSF shall allow the following operations to be associated with SSL/TLS Inspection Proxy rules: block, bypass, or inspect, with the capability to log the operation.

FDP_TEP_EXT.1.4

The TSF shall be able to define monitored clients, requested servers, and [**selection**: *specific client-server connections*, *no other abstractions*] in terms of the attributes associated with the SSL/TLS Inspection Proxy function.

Application Note: This element requires that there must be a mechanism to define a "monitored client" and "requested server" via the attributes specified in FDP_TEP_EXT.1.2. This entity will then have associated rules defined in other elements in this component related to the STIP functionality and operations. If the TOE is able to define a set of attributes that represent a unique client-server connection, then the first selection item should be chosen.

FDP_TEP_EXT.1.5

The TSF shall be able to associate a monitored client, requested server, and [selection: specific client-server connections, no other abstractions] with the allowed TLS version or versions, TLS cipher suites (including TLS key exchange algorithms and key sizes), the supported groups per FCS_TTTC_EXT.5.1, and **[selection: *mutual authentication block-bypass, requested server certificate revocation status unavailable, critical extension in a certificate unrecognized, nothing else*]** that shall be used when performing the SSL/TLS Inspection Proxy operations.

Application Note: This element requires a mechanism that defines, for each "monitored client" and "requested server" (and, if supported, unique client-server pairs), the allowed set of the indicated TLS characteristics associated with those entities. This association allows the enforcement of rules defined by other elements in this component. The first selection is chosen if the TOE supports rules based on both a monitored client and requested server pair.

The second selection indicates events specified in other requirements that need to be associated with monitored clients/requested servers in rules so that appropriate actions can be taken.

The first item is chosen if the TOE supports multiple responses to a client certificate request message from a requested server; FDP_TEP_EXT.1.7 and its Application Note have additional details.

The second item is chosen if FIA_X509_EXT.2 indicates a privileged user may indicate their choice on whether to accept a requested server certificate for which revocation information is not available using allowances. See also FDP_TEP_EXT.1.8.

The third item is chosen if the TOE supports detection of a critical extension in a certificate (being validated according to FIA_X509_EXT.1/STIP) that it cannot interpret. RFC 5280 indicates that this situation results in an invalid certificate, but FIA_X509_EXT.1/STIP provides an additional option that—instead of treating the certificate as invalid (and thus blocking the connection)—the administrator can indicate that the “Bypass” operation is to be applied to the connection instead (which essentially defers the decision to make the connection to the client). See also FDP_TEP_EXT.1.8.

FDP_TEP_EXT.1.6

The TSF shall allow the SSL/TLS Inspection Proxy rules to be assigned to each distinct network interface.

FDP_TEP_EXT.1.7

The TSF shall [**selection**:

- *perform a [**selection**: block, bypass, mutual authentication inspection] operation ,*
- *send an empty certificate list as part of the inspection operation*

] on the session when receiving a TLS certificate request message from the requested server when establishing the TLS in accordance with FCS_TTTC_EXT.1.

Application Note: The ST author will select one or more response options according to the capabilities of the TSF. A mutual authentication inspection operation is a variant of the inspection operation. If this item is selected, the mutual authentication SFRs FCS_TTTC_EXT.3 and FCS_TTTS_EXT.3 must be claimed. If mutual authentication is not supported, one or more of the remaining options is selected: 'Block' and 'send an empty certificate list as part of the inspection operation' are alternative methods to ensure that certificates issued by the TOE's embedded certificate authority are not provided to requested servers that are not known to trust the CA. Block is initiated by the TSF; the TOE terminates the TLS session, whereas 'send an empty certificate list...' allows the requested server to continue with the TLS session without client authentication or terminate the session.

Inspection of mutual authenticated TLS requires both the client and server to trust the embedded CA, and therefore has limited use. It is preferred that inspection of mutual authenticated TLS be performed by components of the requested server security architecture (e.g. via a traffic filtering firewall or an attribute-based access control mechanism) and not be performed by devices described in this PP-Module. If mutual authentication inspection is selected, then the selection-based requirements FCS_TTTC_EXT.3 and FCS_TTTS_EXT.3 will be included by the ST author, and the “mutual authentication” item will be selected in FCS_TTTC_EXT.1.1 and FCS_TTTS_EXT.1.1.

If more than one response option is selected, the ‘mutual authentication block-bypass’ exception specification must be claimed in FDP_TEP_EXT.1.5 and be configurable within the TLS session establishment policy to determine which of the supported operations will be applied for a specific requested server. It is expected, but not required, that one of the selected operations will be a default operation and the other determined by the server matching the exception specification.

FDP_TEP_EXT.1.8

The TSF shall

- Block the connection if the monitored client does not support a TLS version, cipher suite, key exchange, and key size that are in its allowed set as defined in FDP_TEP_EXT.1.5
- Block the connection if the requested server does not negotiate a TLS version, cipher suite, key exchange, and key size that are in its allowed set as defined in FDP_TEP_EXT.1.5
- Either block or [**selection**: *require administrative approval to inspect or bypass*, *no other rule*] the connection if the requested server does not negotiate a TLS version, cipher suite, key exchange, and key size that are in the set proposed by the monitored client in its Client Hello message
- Block or [**selection**: *inspect*, *bypass*, *no other rule*] the connection if TOE certificate processing indicates revocation information is not available for a requested server or [**selection**: *monitored client*, *no other entity*]
- Block or [**selection**: *bypass*, *no other rule*] a connection if TOE certificate

processing indicates an uninterpretable critical extension is present in the certificate of a requested server.

Application Note: Support by a client for the revocation information unavailable case is determined by the TLS handshake protocol messages and fatal errors from the client received during TLS session negotiation with the TOE in accordance with FCS_TTTS_EXT.1.

In the case where a critical extension is encountered that cannot be interpreted by the TOE in accordance with FIA_X509_EXT.1.1/STIP "bypass" can be selected in the last bullet item above. Note that it is not allowed for the administrator to select "Inspect" in this case.

FDP_TEP_EXT.1.9

The TSF shall enforce the following default SSL/TLS Inspection Proxy rules on all SSL/TLS network traffic received from interfaces associated with monitored clients and requested servers:

- The TSF shall drop and be capable of [**selection**: *counting*, *logging*] invalid TLS messages
- The TSF shall drop and be capable of logging TLS Client Hello messages for which no valid client can be determined
- The TSF shall drop a TLS Client Hello message for which no valid server attribute can be determined
- The TSF shall drop and be capable of [**selection**: *counting*, *logging*] TLS messages other than a Client Hello if the message is not associated with an existing TLS session thread established via the inspection operation or a TLS encrypted data flow established via a bypass operation
- The TSF shall terminate a TLS session thread if it receives a fatal TLS error message from the monitored client
- The TSF shall attempt to **[selection:**
  - ***resume the session,***
  - ***renegotiate the session,***
  - ***terminate the TLS session thread and provide a [selection: TLS error message, [assignment: method of notification] to the monitored client associated with the TLS session thread]***

  ] if it receives a fatal TLS error message on the TLS session to the requested server
- The TSF shall terminate a TLS session thread established via the inspect operation, and terminate a TLS encrypted data flow established by the bypass operation, if the TSF receives no traffic from the associated monitored client for a configurable period
- The TSF shall transition a TLS session thread state from inspect operation to block operation, when indicated to do so by the TLS plaintext processing policy

.

Application Note: Dropping a message, performing a block operation, and transitioning to a block operation are different. Dropping a message is typically a silent operation; performing block operation may require messages to be sent to the monitored client associated to the TLS Client Hello; transitioning to a block operation involves termination of the TLS session thread, and potentially sending TLS alert messages to the requested server and TLS alert messages or other messages to the monitored client.

FDP_TEP_EXT.1.10

The TSF shall block all connections for which an Inspection or Bypass operation is not defined.

Application Note: This is the deny by default rule. Note that the block rule does not need to be explicitly defined. This element should not be interpreted that all Client Hello packets should be blocked; the intent is that the Client Hello is initiated from the monitored client, and then the TOE performs processing to determine what to do with the requested connection. If it cannot find a rule that applies for the requested connection, then this element requires that the connection be blocked.

5.2.5 Identification and Authentication (FIA)

FIA_ENR_EXT.1 Certificate Enrollment

FIA_ENR_EXT.1.1

The TSF shall be able to generate a certificate request to an external certification authority to receive a certificate for the TOE's embedded CA's signing key using [**selection**:

- *PKCS#10 in accordance with FIA_X509_EXT.3*,
- *Enrollment over Secure Transport (EST) in accordance with FIA_ESTC_EXT.1*

].

Application Note: The external certification authority may be a root or intermediate certification authority that is used to issue and manage the TOE's embedded CA's certificate. It is not to be used to directly issue end entity certificates to requested servers instead of the TOE's embedded CA.

FIA_X509_EXT.1/STIP X.509 Certificate Validation (STIP)

FIA_X509_EXT.1.1/STIP

The TSF shall validate certificates used for connections supporting STIP functions in accordance with the following rules:

- RFC 5280 certificate validation and certification path validation supporting a minimum path length of three certificates.
- The certification path must terminate with a trusted CA certificate designated as a trust anchor.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.
- The TSF shall validate the revocation status of the certificate using [**selection**: *the Online Certificate Status Protocol (OCSP) as specified in RFC 6960*, *a Certificate Revocation List (CRL) as specified in RFC 5280 Section 6.3*, *Certificate Revocation List (CRL) as specified in RFC 5759 Section 5*].
- The TSF shall validate the extendedKeyUsage field according to the following rules depending on the certificate type and purpose:
  - Server certificates presented in a TLS certificate message for ThruTraffic processing TLS shall have meet one of the following checks:
    - There is no extendedKeyUsage field
    - The extendedKeyUsage field is present and contains the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1)
    - The extendedKeyUsage field is present and contains the 'any' purpose (id-...)
  - Server certificates presented for TLS not associated with the ThruTraffic processing include an extendedKeyUsage field that contains the ServerAuthentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1).
  - Code-signing certificates include the extendedKeyUsage field that contains the CodeSigning purpose.
  - Client certificates presented for TLS for any purpose shall include the extendedKeyUsage field that contains the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.
  - OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.
  - All other certificates used for any other purpose include an extendedKeyUsage field that DOES NOT contain the 'any' purpose.
- The TSF shall validate all extensions marked as critical and verify the value is appropriate for the functionality that uses the value.

FIA_X509_EXT.1.2/STIP

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

Application Note: FIA_X509_EXT.1.1/STIP lists the rules for validating certificates for STIP Functions. The text that says what to do if revocation information is not available, or if a critical extension cannot be processed, is provided in FCS_TTTC_EXT.1.3.

The ST author selects whether revocation status is verified using OCSP or CRLs. The SFR indicates that the TOE be capable of supporting a minimum path length of three certificates. This means that the TOE supports a hierarchy comprising of at least a self-signed root CA certificate, a subordinate CA certificate, and a leaf certificate. The chain validation is expected to terminate with a trust anchor. This means the validation can terminate with any trusted CA certificate designated as a trust anchor. This CA certificate must be loaded into the trust store ('certificate store', ' trusted CA Key Store' or similar) managed by the TOE trust store. If the TOE's trust store supports loading of multiple hierarchical CA

certificates or certificate chains, the TOE must clearly indicate all certificates that it considers trust anchors. The validation of X.509v3 leaf certificates comprises several steps:

a. A Certificate Revocation Check refers to the process of determining the current revocation status of an otherwise structurally valid certificate. This must be performed every time a certificate is used for authentication. This check must be performed for each certificate in the chain up to, but not including, the trust anchor. This means that CA certificates that are not trust anchors, and leaf certificates in the chain, must be checked. It is not required to check the revocation status of any CA certificate designated a trust anchor, however if such check is performed it must be handled consistently with how other certificates are checked.
b. An expiration check must be performed. This check must be conducted for each certificate in the chain, up to and including the trust anchor.
c. The continuity of the chain must be checked, showing that the signature on each certificate that is presented to the TOE is valid and the chain terminates at the trust anchor.
d. The presence of relevant extensions in each certificate in the chain such as the extendedKeyUsage parameters of the leaf certificate must correspond to SFR-relevant functionality. For example, a peer acting as a web server should have TLS Web Server Authentication listed as an extendedKeyUsage parameter of its X.509v3 certificate. The TOE ensuresthat the relevant extensions in each certificate in the chain such as the extendedKeyUsage parameters of the leaf certificate correspond to the SFR-relevant functionality they are used with.

It is expected that revocation checking is performed when a certificate is used in an authentication step. It is expected that revocation checking is performed on both leaf and intermediate CA certificates when a leaf certificate is presented to the TOE as part of the certificate chain during authentication. Revocation checking of any CA certificate designated a trust anchor is not required.

If the TOE implements mutual authentication or acts as a server, there is no expectation of performing any checks on TOE's own leaf certificate during authentication. FIA_X509_EXT.1.2/STIP applies to certificates that are used and processed by the TSF and restricts the certificates that may be added as trusted CA certificates.

5.2.6 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.

5.2.7 Protection of the TSF (FPT)

FPT_FLS.1 Failure with Preservation of Secure State

FPT_FLS.1.1

The TSF shall preserve a secure state when the following types of failures occur: **DRBG failure, integrity test failure, external audit server is unavailable, [selection:** ***local audit storage is full, update signature verification failure, integrity failure on local audit, integrity failure on Trust Anchor database, [assignment: other potential TSF failures]*** **]**.

Application Note: The intent of this requirement is to prevent the use of failed randomization and other events that can compromise the operation of the TOE. This means that the TOE must be able to attain a secure/safe state when any of the identified failures occurs. If the TOE should encounter a failure in the middle of a critical operation, the TOE should not just quit operating, leaving key material and user data unprotected.

The failure of an Operational Environment component can be just as detrimental to security as a failure of the TSF itself. Therefore, in addition to describing the potential TSF failures and how the TOE preserves a secure state in response, the ST author is also expected to use this SFR to express how the TOE is made aware of any environmental failures and how it responds to these

FPT_KST_EXT.1 No Plaintext Key Export

FPT_KST_EXT.1.1

The TSF shall prevent the plaintext export of [**assignment**: *list of all keys used by the TSF*].

Application Note: Keys include all TOE secret and private keys which includes keys generated for issued certificates. The intent of this requirement is to prevent the keys from being exported, even by a security administrator.

FPT_KST_EXT.2 TSF Key Protection

FPT_KST_EXT.2.1

The TSF shall prevent unauthorized use of all TSF private and secret keys.

Application Note: The intent of this requirement is to protect TSF private and secret keys from both unauthorized users, privileged users, and unprivileged processes. Keys specific to the TOE that should be addressed in this requirement include, but are not limited to, the TOE's embedded CA's private signing key, private keys associated to certificates issued by the TOE's embedded CA and TLS session keys established to facilitate inspection of traffic. Users should not be able to access the keys through "normal" interfaces. Processes that use private or secret keys to meet the functionality described in this PP module are considered authorized; all other processes are unauthorized. When an interface allows both authorized and unauthorized access to a key (for example, certificate signing functions with access to the embedded CA's private signing key are authorized only when certificate to be signed corresponds to a valid certificate belonging to a requested, and is unauthorized at any other time), evidence of protection includes logging of accesses via the common interface, as indicated in Table 2 for FAU_GEN.1.

FPT_RCV.1 Manual Trusted Recovery

FPT_RCV.1.1

After [**assignment**: *list of failures/service discontinuities*] the TSF shall enter a maintenance mode where the ability to return to a secure state is provided.

Application Note: This requirement ensures that the TSF can determine that the TOE is started up without protection compromise and can recover without protection compromise after discontinuity of operations. Anticipated failures include actions that result in a system crash, media failures, or discontinuity of operations caused by erroneous administrative action or lack of erroneous administrative action. The data that needs to be restored includes the TSF keys needed for signature, the Trust Anchor Database, keys needed for management of certificates, all signed certificates, and any certificate status information

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

FAU_GCR_EXT.1
FAU_STG.4
FCS_COP.1/STIP
FCS_STG_EXT.1
FCS_TTTC_EXT.1
FCS_TTTC_EXT.5
FCS_TTTS_EXT.1
FDP_CER_EXT.1
FDP_CER_EXT.2
FDP_CER_EXT.3
FDP_CSIR_EXT.1
FDP_PPP_EXT.1
FDP_PRC_EXT.1
FDP_RIP.1
FDP_STG_EXT.1
FDP_STIP_EXT.1
FDP_TEP_EXT.1
FIA_ENR_EXT.1
FIA_X509_EXT.1/STIP
FPT_FLS.1
FPT_KST_EXT.1
FPT_KST_EXT.2
FPT_RCV.1

Optional SFRs

FAU_SAR.1
FAU_SAR.3
FDP_PIN_EXT.1

Selection-based SFRs

FDP_CRL_EXT.1
FDP_CSI_EXT.1
FDP_OCSP_EXT.1
FCS_OCSPS_EXT.1
FIA_ESTC_EXT.1

FTA_TAB.1/TLS	
FCS_TTTC_EXT.3	
FCS_TTTS_EXT.3	
FDP_CER_EXT.4	
FDP_CER_EXT.5	
FDP_CSI_EXT.2	
FDP_STIP_EXT.2	
FAU_SCR_EXT.1	
FCS_CKM_EXT.5	
FCS_TTTC_EXT.4	
FCS_TTTS_EXT.4	
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	
This PP-Module does not define any Objective requirements.	
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 Security Assurance requirements. The SARs from the Base-PP must be satisfied.

Appendix A - Optional SFRs

A.1 Strictly Optional Requirements

A.1.1 Persistent Local Audit Storage

The SFRs in this section are optional. They should be claimed if the TOE provides local audit storage (i.e., if FAU_STG.1 is claimed in the base NDcPP) and that local audit storage is intended to provide a persistent, searchable record of security events within the TOE, either as a backup or replacement of an external audit capability.

FAU_SAR.1 Audit Review

FAU_SAR.1.1

The TSF shall provide [**selection**: *Security Administrators*, *Auditors*] with the capability to read all information from the local audit records

FAU_SAR.1.2

The TSF shall provide the local audit records in a manner suitable for the administrator to interpret the information.

FAU_SAR.3 Selectable Audit Review

FAU_SAR.3.1

The TSF shall provide the ability to apply searches of local audit data based on **[assignment: *object identifier of certificate*] associated with the event**.

Application Note:

A.1.2 Certificate Pinning

Certificate pinning is an optional feature to address the threat of unauthorized access to user data managed by the TOE via unauthorized STIP or adversary man-in-the-middle exploits. This feature is desirable since implementation of a STIP to protect a client enclave will prevent the clients from effectively providing this feature.

FDP_PIN_EXT.1 Certificate Pinning

FDP_PIN_EXT.1.1

The TSF shall be able to detect and [**selection**: *alert*, *[**assignment**: perform a [Security Administrator] managed action]*] to changes in the [**selection**: *public key*, *certificate*, *certificate issuer*] used by requested servers according to **[selection: *a Security Administrator configurable number of the most common requested servers, a Security Administrator specified list of servers, [assignment: a Security Administrator configurable rules based on attributes of the certificates used by the server]*]** .

Application Note: This requirement should be claimed if implemented by the TOE. If claimed, additional FMT_MOF.1 and audit events associated with the function must be claimed.

A.2 Objective Requirements

This PP-Module does not define any Objective SFRs.

A.2.1 Identification and Authentication (FIA)

A.3 Implementation-Based Requirements

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

Appendix B - Selection-Based Requirements

B.1 Certificate Status Information

FDP_CRL_EXT.1 Certificate Revocation List Generation

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

FDP_CRL_EXT.1.1

When the TSF is configured to generate CRLs, the TSF shall verify that all mandatory fields in any generated CRL contains values in accordance with ITU-T Recommendation X.509. At a minimum, the following items shall be validated:

a. If the version field is present, then it shall contain a 1.
b. If the CRL contains any critical extensions, then the version field shall be present and contain the integer 1.
c. The [**selection**: *issuer*, *issuerAltName*] fields shall indicate the configured name of the CA.
d. If the issuer field contains a null Name (e.g., a sequence of zero relative distinguished names), then the CRL shall contain a critical issuerAltName extension.
e. The signature and signatureAlgorithm fields shall contain the OID for a digital signature algorithm in accordance with FCS_COP.1**/SigGen in the NDcPP**.
f. The thisUpdate field shall indicate the issue date of the CRL.
g. The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.

Application Note: This requirement should be claimed if 'ITU-T Recommendation X.509v2 CRL' is selected in FDP_CSI_EXT.1.1

FDP_CSI_EXT.1 Certificate Status Information

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

FDP_CSI_EXT.1.1

The TSF shall generate certificate status information whose format complies with [**selection**: *ITU-T Recommendation X.509v2 CRL, the OCSP standard as defined by RFC 6960*].

FDP_CSI_EXT.1.2

The TSF shall support changes to the status of a certificate by [**selection**:

- *[**selection**: Security Administrator, CA Operations Staff] ,*
- *[**assignment**: automated revocation rules]*

].

FDP_CSI_EXT.1.3

The TSF shall [**selection**: *provide*, *interface with the Operational Environment to provide*] certificate status information generated in accordance with FDP_CSI_EXT.1.1 via [**selection**: *posting CRLs at the location specified in the cRLDistributionPoints of the issued certificate*, *an OCSP mechanism indicated in the authorityInfoAccess extension of the issued certificate*, *OCSP Stapling in accordance with FDP_OCSPS_EXT.1*].

Application Note: This SFR should be claimed if claimed if the selection 'generate certificate status information' is selected in FDP_CSIR_EXT.1.1.

The ST should specify the format(s) used to supply certificate status information in FDP_CSI_EXT.1.1, and the mechanism(s) used to provide the status to relying parties including all monitored clients in the second selection in FDP_CSI_EXT.1.3. If CRLs are identified in FDP_CSI_EXT.1.1, then cRLDistributionPoints must be claimed in FDP_CSI_EXT.1.3 and in FDP_CER_EXT.1.2 item (g), sub-item (g). If the OCSP standard is selected in FDP_CSI_EXT.1.1, then at least one of the last two options in the second selection of FDP_CSI_EXT.1.3 must be claimed. If the second option (OCSP) is claimed, authorityInfoAccess must be claimed in FDP_CER_EXT.1.2 item (g), sub-item (g). OCSP stapling may also be claimed if the TOE only generates CRLs, but interfaces with an external OCSP responder that uses those CRLs.

Automated rules for revoking certificates in response to the TOE's discovery that

a previously issued certificate is no longer appropriate for the subject, or due to cache clearing, timeouts, or other rules should be described in the assignment of FDP_CSI_EXT.1.2.

FDP_OCSP_EXT.1 OCSP Basic Response Generation

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

FDP_OCSP_EXT.1.1

When the TSF is configured to generate OCSP responses of the basic response type, the TSF shall ensure that all mandatory fields in the OCSP basic response contain values in accordance with RFC 6960. At a minimum, the following items shall be validated:

a. The version field shall contain a 0.
b. The signatureAlgorithm field shall contain the object identifier (OID) for a digital signature algorithm in accordance with FCS_COP.1**/SigGen in the NDcPP**.
c. The thisUpdate field shall indicate the time at which the status being indicated is known to be correct.
d. The producedAt field shall indicate the time at which the OCSP responder signed the response.
e. The time specified in the nextUpdate field (if populated) shall not precede the time specified in the thisUpdate field.

Application Note: This requirement should be claimed if 'the OCSP standard as defined by RFC 6960' is selected in FDP_CSI_EXT.1.1.

FCS_OCSPS_EXT.1 OCSP Stapling

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

FCS_OCSPS_EXT.1.1

The TSF shall be able to process [**selection**: *Certificate Status Request extension in accordance with RFC 6066 section 8*, *Certificate Status Request List V2 in accordance with RFC 6961*].

FCS_OCSPS_EXT.1.2

The TSF shall [**selection**: *generate OCSP response information in accordance with FDP_OCSP_EXT.1*, *interface with an OCSP provider to obtain an OCSP response*] and populate a Certificate Status Message in accordance with RFC 6066.

Application Note: This SFR must be claimed in situations where the TOE computes OCSP responses for inclusion in TLS certificate status messages. It may also be claimed if the TOE includes a separate certificate status component (CRL or OCSP provider) and provides an interface to an internal or external OCSP responder that processes certificate status information provided to it by the TOE. When claimed, certificate status is provided via OCSP stapling contained within TLS Server certificate status message(s).

B.2 Certificate Enrollment

FIA_ESTC_EXT.1 Enrollment over Secure Transport (EST) Client

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

FIA_ESTC_EXT.1.1

The TSF shall use the Enrollment over Secure Transport (EST) as specified in RFC 7030 to obtain its embedded CA certificate and [**assignment**: *other certificates for the TOE*] from an external certification authority (external CA) associated with an authorized EST server.

FIA_ESTC_EXT.1.2

The TSF shall be able to obtain EST server and CA certificates for authorized EST services via [**selection**:

- *implicit Trust Anchor/Trust Store (TA) configured by* ***[selection: Security Administrator, CA Operations Staff]*** ,
- *an explicit TA populated via a TLS-authenticated EST CA certificate request in accordance with RFC 7030 section 4.1.2 and FCS_TLSC_EXT.1*

].

FIA_ESTC_EXT.1.3

The TSF shall authenticate EST servers using X.509 certificates that chain to trust store elements from the [**selection**: *implicit Trust Anchor database*, *explicit Trust Anchor/Trust Store*] in accordance with FIA_X509_EXT.1**/Rev** for all EST requests.

FIA_ESTC_EXT.1.4

The TSF shall authenticate its certificate enrollment requests to receive the signing certificate of its embedded CA and [**assignment**: *other certificates required to authenticate the TOE*], from an authorized EST server using [**selection**:

- *HTTP basic authentication transported over TLS in accordance with RFC 7030 section 3.2.3 and FCS_TLSC_EXT.1*,
- *HTTP digest authentication using a cryptographic hash algorithm in accordance with FCS_COP.1**/Hash**, transported over TLS in accordance with RFC 7030 section 3.2.3 and FCS_TLSC_EXT.1*,
- *Certificate-based authentication in accordance with RFC 7030 section 3.3.2 and FCS_TLSC_EXT.2 using [**assignment**: a pre-existing certificate authorized by the EST server]*

].

FIA_ESTC_EXT.1.5

The TSF shall generate authenticated re-enrollment requests in accordance with RFC 7030 Section 3.3.2 and FCS_TLSC_EXT.1, using an existing valid certificate with the same subject name as the requested certificate and which was issued by the external CA.

Application Note: This SFR should be claimed if 'Enrollment over Secure Transport…' is claimed in FIA_ENR_EXT.1.1.

B.3 Inspection Policy Banner

Local policy may require explicit consent to monitoring before inspection of TLS encrypted data. If the STIP may be deployed in an environment where clients might not already have granted this approval, the TOE might be required to obtain this consent. The requirement in this section should be claimed if the TLS session establishment policy requires it.

FTA_TAB.1/TLS TOE Access Banner (Consent to Monitor Banner for TLS Inspection

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

FTA_TAB.1.1/TLS

Before forwarding decrypted application data intended for the requested server to inspection processing components the TSF shall display **to the monitored client a Security Administrator-specified** advisory **notice and consent** warning message regarding use of the TOE.

Application Note: This SFR should be claimed if 'provide a consent to monitor banner..' is selected in FDP_STIP_EXT.1.3.

B.4 Authentication of Monitored Clients

This section describes support for mutual authentication of clients when requested by the TOE to support the following use cases:

- TOE requested mutual authentication: Mutual authentication provides authenticated client attributes that can be used to define exception processing. If the ST claims to support client authentication of monitored clients accessing the TOE, this SFR should be claimed in the selection of FDP_STIP_EXT.1.
- Certificate request from the requested server: Inspection of TLS sessions requiring mutual authentication is a narrow use case for the SSL/TLS inspection proxy, where both the client and server trust the TOE's embedded CA. In such instances, mutual authentication represents an assertion to the requested server that the client has been authenticated. There are other, preferred mechanisms such as SOAP, KERBEROS, XAML assertions, than TLS client authentication using proxy certificates that provide a more accurate representation of the role of a federated identity service provider, in accordance with NIST SP 800-63-03. Also, mutual authentication is typically used to control access to sensitive information authorized only to specific clients, and the willingness of the server's content owner to trust the proxy, providing access such sensitive content further restricts legitimate use cases. Finally, certificates issued to represent users subject to a certificate policy or certificate practice statement, especially those compliant with NIST FIPS 201, may be required to meet an equivalent certificate policy or certificate practice statement.

If mutual authentication by the TOE is to perform the TLS inspection operation on TLS sessions between monitored clients and requested servers requiring mutual authentication, FCS_TTTC_EXT.3, FCS_TTTS_EXT.3, FDP_CER_EXT.4, FDP_CER_EXT.5, FDP_CSI_EXT.2, and FDP_STIP_EXT.2 in this section must be claimed. In addition, the 'mutual authentication inspection' item should be selected in the selection for FDP_TEP_EXT.1.5 and an exception specification to identify servers which are authorized and configured to support mutual authentication inspection must be described in the assignment of FDP_TEP_EXT.1.4. TLS servers requesting mutual authentication are likely to also require revocation information, so it is recommended that FDP_CSIR_EXT.1 selections be made to provide certificate status information, even if the constraint for short validity periods is achieved

FCS_TTTC_EXT.3 Thru-Traffic TLS Inspection Client Protocol with Mutual Authentication Representing Monitored Clients

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

FCS_TTTC_EXT.3.1

The TSF shall support mutual authentication using X.509v3 certificates generated in accordance with FDP_CER_EXT.5 for inspection processing operation between a monitored client represented in the generated certificate and a requested server that provides a certificate request in the TLS handshake.

Application Note: This SFR must be claimed if the TSF is capable of inspecting TLS sessions from monitored clients to requested servers requiring client authentication.

FCS_TTTS_EXT.3 Thru-Traffic TLS Inspection Server Protocol with Mutual Authentication of Monitored Clients

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

FCS_TTTS_EXT.3.1

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

FCS_TTTS_EXT.3.2

The TSF shall send a Certificate Request message to the TLS client when mutual authentication is required by the configured TLS session establishment policy as defined in FDP_TEP_EXT.1.

FCS_TTTS_EXT.3.3

The TSF shall validate the certificate presented by the client and [**selection**: *allow the connection if the certificate is invalid and an exception is permitted for the client, terminate the connection if the certificate is invalid*].

Application Note: Validity is determined by the identifier verification, certificate path, the expiration date, and the revocation status in accordance with RFC 5280. Certificate validity is tested in accordance with testing performed for FIA_X509_EXT.1/STIP

FCS_TTTS_EXT.3.4

The TSF shall not establish a TLS session if the distinguished name (DN) or Subject Alternative Name (SAN) contained in a certificate does not match the expected identifier for the client.

Application Note: The client identifier may be in the Subject field or the Subject Alternative Name extension of the certificate.

FDP_CER_EXT.4 Certificate Profiles for Client Certificates

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

FDP_CER_EXT.4.1

The TSF shall implement a certificate profile function for TLS client certificates issued by a CA embedded within the TOE, and shall ensure that issued certificates are consistent with configured profiles.

Application Note: The CA issuing client certificates may be required to support configured certificate profiles that differ significantly from server certificates, and to support multiple certificate profiles for the clients supported.

FDP_CER_EXT.4.2

The TSF shall generate certificates representing monitored clients using profiles that comply with requirements for certificates as specified in IETF RFC 5280,

“Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile” as refined below. At a minimum, the TSF shall ensure that:

a. The version field shall contain the integer 2.
b. The issuerUniqueID or subjectUniqueID fields are not populated.
c. The serialNumber shall be unique with respect to the issuing Certification Authority.
d. The validity field shall specify a notBefore value that does not precede the current time and a notAfter value that does not precede the value specified in notBefore.
e. The issuer field is not empty and is populated with the **[selection: *Security Administrator, CA Operations Staff*]**-configured CA name.
f. The signature field and the algorithm in the subjectPublicKeyInfo field shall contain the OID for a signature algorithm specified in FCS_COP.1/SigGen in the NDcPP.
g. The following extensions are supported:
    a. subjectKeyIdentifier
    b. authorityKeyIdentifier
    c. keyUsage
    d. extendedKeyUsage
    e. certificatePolicy
    f. [**selection**: *basicConstraints*, *cRLDistributionPoints*, *authorityInfoAccess*, *no other extensions*]
h. A subject field containing a null Name (e.g., a sequence of zero relative distinguished names) is accompanied by a populated critical subjectAltName extension.
i. The authorityKeyIdentifier extension in any certificate issued by the TOE must be populated and must be the same as the subjectKeyIdentifier extension contained in the TOE’s embedded CA’s signing certificate
j. Populated keyUsage and extendedKeyUsage fields in the same certificate shall contain consistent values reflecting exclusive TLS server use as follows:

keyUsage	extendedKeyUsage
digitalSignature	clientAuth
digitalSignature, keyEncipherment	clientAuth
digitalSignature,keyAgreement	clientAuth

Application Note: RFC updates to RFC 5280 are included in this requirement. The inclusion of the cRLDistributionPoints and authorityInfoAccess extensions depend on the selections made in FDP_CSIR_EXT.1 and FDP_CSI_EXT.2.3 if claimed.

Uniqueness for the subject key identifier is specific to the instance of the embedded CA. The same configured CA should not issue certificates with different public keys having the same subject key identifier.

FDP_CER_EXT.4.3

The TSF shall implement the following rules for populating certificate fields based on constraints imposed by the TOE’s embedded CA’s signing certificate:

- The validity field shall specify a notAfter time that does not exceed the notAfter time of the CA’s signing certificate.
- The issuer field identifies the [**selection**:
    - *subject*,
    - *[**assignment**: **[selection: Security Administrator, CA Operations Staff]**-assigned identifying information]*

  ] of the CA's signing certificate.
- [**selection**:
    - *The subject name is limited by name constraints specified in the CA’s signing certificate*,
    - *[**assignment**: list of rules]*,
    - *no other rules*

  ]

FDP_CER_EXT.4.4

The TSF shall implement the following rules for populating certificate fields based on the validated certificate and constraints imposed by the **[selection: *Security Administrator, CA Operations Staff*]**:

a. The Subject/Subject Alternative Name shall be copied from validated client

certificate.

b. The notBefore field shall not precede the notBefore field of the validated client certificate.
c. The notAfter field shall not exceed the notAfter field of the validated client certificate.
d. The notAfter field shall not exceed the current time by more than a maximum validity duration value as configured by a **[selection: *Security Administrator, CA Operations Staff*]** user.
e. If the basicConstraints field is configured to be present, it shall be populated with the value cA=False
f. If configured to be present, th epolicy OID and policy mapping fields shall be populated according to [**selection**:
    - *a **[selection: Security Administrator, CA Operations Staff]** configured mapping from validated client certificate values to one or more stated policy OIDs,*
    - *[**assignment**: list of rules]*

]

Application Note: Policy OIDs for proxy-issued certificates and mappings to FIPS 201 defined policies may be required to be supported for SSL/TLS inspection proxies issuing certificates representing person-entities subject to FIPS 201 authentication methods, since requested servers requiring client authentication are likely to expect to validate client certificates issued by the equivalent of such a certificate policy. If Policy OIDs are used, the embedded CA may be subject to additional constraints indicated in a Certificate Policy.

FDP_CER_EXT.5 Certificate Issuance Rules for Client Certificates

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

FDP_CER_EXT.5.1

The TSF shall issue certificates in response to a validated client certificate according to the following rules: The issued certificate is in compliance with a current certificate profile defined in accordance with FDP_CER_EXT.5 and

- The TLS session establishment policy is configured to allow inspection of TLS sessions with mutual authentication between the monitored client whose certificate is validated by the TSF and one or more requested servers,
- The specific requested server includes a Certificate Request message in the TLS handshake,

[**selection**:

- *A valid certificate for the same subject is not present in cache,*
- *The embedded CA certificate's name space allows issuance of a certificate that represents the authenticated client,*
- *No other constraints*

].

Application Note: Caching client certificates is neither required nor preferred, since such storage of private keys associated to signature keys is strictly controlled by various certificate policies and practice statements, especially when the validated client certificate associated to the monitored client is issued under NIST FIPS 201. If supported, the time in cache for client certificates should be limited to the minimal revocation time (emergency revocation) allowed for validated certificates used by any monitored client.

FDP_CER_EXT.5.2

The TSF shall reject all certificate requests originating external to the TOE.

FDP_CSI_EXT.2 Certificate Status Information for Client Certificates

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

FDP_CSI_EXT.2.1

The TSF shall generate certificate status information for issued client certificates whose format complies with [**selection**: *ITU-T Recommendation X.509v2 CRL, the OCSP standard as defined by RFC 6960*].

FDP_CSI_EXT.2.2

The TSF shall support changes to the status of a certificate in accordance with the following rules:

- as directed by **[selection: *Security Administrator, CA Operations Staff*]**

and

- [**selection**:
  - *a certificate in cache is revoked when a certificate representing the same subject is received for client authentication and either*
    - *the validation of the received certificate fails, or*
    - *the validation of the received certificate passes and the certificate fields of the validated certificate would result in a different certificate being issued under the current profile in accordance with FDP_CER_EXT.4*

    ,
  - *[**assignment**: other rules for revocation of issued certificates],*
  - *no other rules*

  ]

Application Note: In order to meet revocation requirements associated with credentials issued under FIPS 201, the first item of the second selection in this element must be claimed if cache is provided. Automated rules for revoking certificates in response to the TOE's discovery that a previously issued certificate is no longer appropriate for the subject, or due to cache clearing, timeouts, or other rules should be described in the assignment of FDP_CSI_EXT.2.2.

FDP_CSI_EXT.2.3

The TSF shall [**selection**: *provide*, *interface with the Operational Environment to provide*] certificate status information generated in accordance with FDP_CSI_EXT.2.1 via [**selection**: *posting CRLs at the location specified in the cRLDistributionPoints of the issued certificate*, *an OCSP mechanism indicated in the authorityInfoAccess extension of the issued certificate*].

Application Note: Based on the selection, the ST author must choose the appropriate requirements from Appendix B.1 of this PP-Module.

The ST should specify the format(s) used to supply certificate status information in FDP_CSI_EXT.2.1, and the mechanism(s) used to provide the status to relying parties including all servers authorized to use mutually authenticated TLS in accordance with the configured TLS session establishment policy, identified in FDP_TEP_EXT.1, in the second selection in FDP_CSI_EXT.2.3. If CRLs are identified in FDP_CSI_EXT.2.1, then cRLDistributionPoints must be claimed in FDP_CSI_EXT.2.3 and in FDP_CER_EXT.4.2 item (g), sub-item (f). If the OCSP standard is selected in FDP_CSI_EXT.2.1, then 'authorityInfoAccess' must be selected in the second selection of FDP_CSI_EXT.2.3 and in FDP_CER_EXT.4.2 item (g), sub-item (f).

FDP_STIP_EXT.2 Mutual Authentication Inspection Operation

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

FDP_STIP_EXT.2.1

The TSF shall be capable of providing mutual authentication of the monitored client to a requested server when performing the inspection operation when mutual authentication is allowed for the requested server by the configured policy, and the TLS handshake with the requested server includes a certificate request.

Application Note: The policy is flexible; it could be static policy or a policy associated with certain clients, servers, or connections.

FDP_STIP_EXT.2.2

After receiving the TLS client certificate from the monitored client, the TSF shall be able to generate a certificate representing the client in accordance with FDP_CER_EXT.5 and [**selection**: *obtain a valid certificate representing the client from cache*, *no other method*] matching the current certificate profile.

Application Note: Certificate caching of client certificates is not required. However, in the case where certificate caching is supported, the TSF will still need to perform certificate generation if the cached certificate does not match the current profile determined by FDP_CER_EXT.4 which depends on values derived from the certificate provided by the monitored client.

FDP_STIP_EXT.2.3

After obtaining a certificate representing the monitored client, the TSF shall send the client certificate and certificate verify messages to the requested server.

Application Note: This element completes the TLS handshake between the TOE and the requested server as a complete TLS handshake with mutual authentication.

B.5 Other Selection-Based SFRs

FAU_SCR_EXT.1 Certificate Repository Review

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

FAU_SCR_EXT.1.1

The TSF shall [**selection**: *provide*, *invoke the Operational Environment to provide*] the ability to search certificates containing specified values of the following certificate fields: **[selection:**

- ***subject name,***
- ***individual components of Subject Alternative Name,***
- ***subject ID,***
- ***issuer ID,***
- ***algorithm ID,***
- ***public key,***
- ***key usage,***
- ***extended key usage,***
- ***serial number,***
- ***[assignment: list of other certificate fields]***

] returning all matching certificates and [**assignment**: *object identifier(s)*] of matching certificate(s).

Application Note: This SFR must be claimed if the selection in FAU_GCR_EXT.1.1 is 'store.' It may be claimed if the selection in FAU_GCR_EXT.1.1 is 'invokes the Operational Environment to store' when the TSF provides an interface to the certificate repository to perform searches. The ability to search on certificate fields is useful for conducting forensic analysis. If the certificate repository is stored within the TOE boundary, then the first item of the first selection is chosen. If the repository is stored in the OE, but the auditor uses TSF interfaces to perform this function on the repository, then the second item of the first selection is chosen. It is allowed that this function be provided entirely by the OE (when the repository is stored in the OE); if this is the case, then this requirement is not included in the ST, but instead the OE.CERTIFICATE_REPOSITORY_SEARCH objective is included (this objective is omitted in the other two cases, when this SFR is included in the ST).

In the second selection and assignment, the ST author includes/fills in the values that can be searched on for this function; at least one value is required to be selected.

FCS_CKM_EXT.5 Public Key Integrity

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

FCS_CKM_EXT.5.1

The TSF shall protect persistent public keys against undetected modification through the use of [**selection**: *digital signatures* ***(in accordance with FCS_COP.1/SigGen)***, *keyed hashes* ***(in accordance with FCS_COP.1/KeyedHash)***].

FCS_CKM_EXT.5.2

The [**selection**: *digital signature*, *keyed hash*] used to protect a public key shall be verified upon [**assignment**: *criteria for automated verification*].

Application Note: This SFR is included when the second selection in FDP_STG_EXT.1.1 is chosen, and applies to the public keys listed in that SFR.

The selections in FCS_CKM_EXT.5.1 and FCS_CKM_EXT.5.2 should agree, and the assignment in FCS_CKM.5.2 for the criteria for automated verification can be event or time based and should provide operationally relevant integrity failure detection, for which recovery is feasible.

FCS_TTTC_EXT.4 STIP Client-Side Support for Renegotiation

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

FCS_TTTC_EXT.4.1

The TSF shall support secure renegotiation on STIP TLS connections through use of the “renegotiation_info” TLS extension in accordance with RFC 5746.

FCS_TTTC_EXT.4.2

The TSF shall include [**selection**: *renegotiation_info extension*, *TLS_EMPTY_RENEGOTIATION_INFO_SCSV cipher suite*] in the Client Hello message.

Application Note: This SFR is included when “session renegotiation” in FCS_TTTC_EXT.1.1 is chosen. RFC 5746 defines an extension to TLS that binds renegotiation handshakes to the cryptography in the original handshake. The cipher suite included in the selection is a means for clients to be compatible with servers that don’t support the extension. It is recommended that client implementations support both the cipher suite and the extension.

FCS_TTTC_EXT.4.3

The TSF shall ensure that renegotiation is performed before [**selection**: *[**assignment**: renegotiation rules], 2^20 64-bit data blocks are encrypted using TDES cipher suites using the same key*].

Application Note: If a TDES cipher suite is selected in FCS_TTTC_EXT.1.1, the amount of data encrypted with the same key is limited in accordance with NIST SP800-67R2, section 3.4, and the second selection should be chosen.

FCS_TTTS_EXT.4 STIP Server-Side Support for Renegotiation

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

FCS_TTTS_EXT.4.1

The TSF shall support the "renegotiation_info" TLS extension in accordance with RFC 5746.

FCS_TTTS_EXT.4.2

The TSF shall include the renegotiation_info extension in Server Hello messages.

Application Note: This SFR is included when “session renegotiation” in FCS_TTTS_EXT.1.1 is chosen. RFC 5746 defines an extension to TLS that binds renegotiation handshakes to the cryptography in the original handshake.

B.6 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 .

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.

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 4: Extended Component Definitions

Functional Class	Functional Components
Identification and Authentication (FIA)	FIA_PSK_EXT Pre-Shared Key Composition

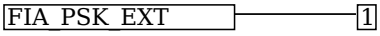
C.2 Extended Component Definitions

C.2.1 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, 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*

].

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 5: 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