PP-Module for Authentication Servers

Version: 1.0
2022-08-12
National Information Assurance Partnership

Revision History

Version	Date	Comment
1.0	2022-08-12	Initial Release

Contents

1 Introduction

1.1 Overview

An authentication server provides assertions to a relying party that a particular request for access is from an authentic digital identity associated with various identity attributes, such as a registered account within an information system, or a certified identity as validated by a trusted certification authority or both. The digital identities can represent people, devices, or processes. Authentication servers validate various authenticators controlled by the entities represented by the presented digital identity. When the entity is a person, authenticators can provide indications of what the entity knows (e.g., a password, pin, or passphrase), what the entity has (e.g., a registered device in the control of the user), or what the entity is (a biometric). NIST SP 800-63-3 Part B provides recommendations about how these authenticators can be leveraged individually or in combinations to provide assurance that the entity is authentic and describes requirements for validation of the authenticators to various assurance levels.

A relying party may delegate verification of authenticators to an authentication server; such delegation creates a relationship between the relying party and the authentication server that is referred to as an identity federation. Assertions to a federated relying party can be via bearer assertions or via direct communication with the relying party. The latter mechanism is modeled after that used by Authentication, Access, and Accounting (AAA) servers, which used the RADIUS protocol. RADIUS has been largely replaced by DIAMETER, a protocol that addresses many of the security issues with RADIUS. These provide direct, back-end assertions protected by an authenticated and encrypted channel to a Network Access Server (NAS) that further governs accesses to resources on a network.

This PP-module describes the security functionality of authentication servers supporting RADIUS or DIAMETER and other messaging protocols intended for direct communications with relying parties via authenticated, real-time protected channels.

The scope of this PP-Module is to describe the security functionality of an authentication server 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-PP:

- collaborative Protection Profile for Network Devices, Version 2.2e (NDcPP)

This Base-PP is valid because an authentication server can be deployed as a dedicated network appliance. The use case of deploying the authentication server as an application on a general-purpose computer is outside the scope of this PP-Module. Authentication server products allow enterprises to provide a centralized and standardized method of evaluating user authentication requests made throughout the enterprise. This enables a centralized definition for user identity and credential data and allows for uniform application of authentication policies that define what credentials and user attributes are necessary to gain access to various systems and applications in the enterprise environment.

Note that the NDcPP defines an optional architecture for a “distributed TOE” that allows for security functionality to be spread across multiple distinct components. This PP-Module does not require or prohibit the TOE from being a distributed system when the TOE conforms to the NDcPP; the TOE may be standalone or distributed in this case.

1.2 Terms

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

1.2.1 Common Criteria Terms

Assurance	Grounds for confidence that a TOE meets the SFRs [CC].
Base Protection Profile (Base-PP)	Protection Profile used as a basis to build a PP-Configuration.
Collaborative Protection Profile (cPP)	A Protection Profile developed by international technical communities and approved by multiple schemes.
Common Criteria (CC)	Common Criteria for Information Technology Security Evaluation (International Standard ISO/IEC 15408).
Common Criteria Testing Laboratory	Within the context of the Common Criteria Evaluation and Validation Scheme (CCEVS), an IT security evaluation facility accredited by the National Voluntary Laboratory Accreditation Program (NVLAP) and approved by the NIAP Validation Body to conduct Common Criteria-based evaluations.
Common Evaluation Methodology	Common Evaluation Methodology for Information Technology Security Evaluation.

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

1.2.2 Technical Terms

Assertion	A statement from the TOE to an RP that contains information about a subscriber. Assertions may also contain verified attributes. For the purposes of this PP-Module, assertions containing authentication status and identity attributes are made by EAP response messages in accordance with EAP-TLS or EAP-TTLS.
Authentication Policy	A policy that specifies which authenticator types are required for a particular entity. The policy may be implicit for all entities, or configurable.
Authenticator	Something the claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the claimant’s identity.
Authenticator Output	The output value generated by an authenticator. The ability to generate valid authenticator outputs on demand proves that the claimant possesses and controls the authenticator. Protocol messages sent to the verifier are dependent upon the authenticator output, but they may or may not explicitly contain it.
Claimant	A subject whose identity is to be verified using one or more authentication protocols.
Credential	An object or data structure that authoritatively binds an identity via an identifier or identifiers and (optionally) additional attributes, to at least one authenticator possessed and controlled by a subscriber.
Federation Protocol	A protocol to establish a trusted relationship with a relying party, and for the purposes of this PP module, to communicate authentication status for entities requesting access to resources managed by the relying party. In this PP-module, Federation Protocols include RADIUS, DIAMETER, and other standard protocols used in direct communication between the relying party and the TOE. Federation protocols that only support bearer assertions are

	out of scope for this PP-Module.
Relying Party (RP)	An entity that relies upon the subscriber's authenticators and credentials or a verifier's assertion of a claimant's identity, typically to process a transaction or grant access to information or a system.

1.3 Compliant Targets of Evaluation

This PP-Module specifically addresses a dedicated network device that performs entity (device or user) authentication via direct, back-end connections with a relying party. The entity to be authenticated is referred to as the claimant, though different terms have been used for specific protocols (e.g., peer for RADIUS or DIAMETER). The relying party can manage a single resource or provide access control for resources within a network. For example, a Wireless Local Area Network (WLAN) Access System may use the services of a dedicated authentication server during tunnel establishment. In this use case, an authentication server must support IEEE 802.1X Port-Based Network Access Control and must fulfill the IEEE 802.11 authentication server role using Extensible Authentication Protocol (EAP) messaging.

Similarly, the authentication server may be used during Virtual Private Network (VPN) tunnel establishment. The relying party in this case is a VPN Gateway acting as a Network Access Server using pass-through between the VPN client and authentication server (the TOE), also using EAP messaging.

In general, any relying party using a direct authentication federation protocol that supports EAP-TLS or EAP-TTLS messaging is addressed by this PP-Module.

The combination of the NDcPP and this PP-Module is a network device that provides authentication server functionality in addition to all of the security functionality expected of a network device as mandated by the NDcPP.

This PP-Module describes the functional requirements and threats specific to authentication servers. A TOE that conforms to this PP-Module must also conform to the Base-PP.

1.4 TOE Boundary

This document specifies SFRs for an authentication server. An authentication server is designed to authenticate a claimant that attempts to access a relying party – an access gateway to a protected network, or individual resources and services – and provide assertions to one or more relying parties about the authentication state of the claimant. A claimant forwards one or more authenticator outputs to the authentication server; the authentication server verifies the authenticator outputs and may also provide additional identity attributes to allow the relying party to determine whether the claimant meets its authentication policy.

The authentication server defined by this PP-Module is one or more dedicated network appliances; the TOE is not intended to run as an application on a general-purpose computer. The authentication server can be co-located with an access management or privilege management system, or it may be separate from such services. Regardless of the deployment, access control functions and management of non-identity attributes are outside the scope of this PP-Module.

An authentication server may be part of a larger system that also provides authorization information, either as part of an AAA server, an authorization server, or a domain controller. This PP-Module specifies the functional requirements for authentication services only; as in the case where the TOE may be co-located with the relying party, the TOE's logical boundary only includes the authentication server functionality. However, the TOE boundary includes the ability to generate audit events that are specific to the authentication functionality but may be used to support other functions (e.g., AAA servers).

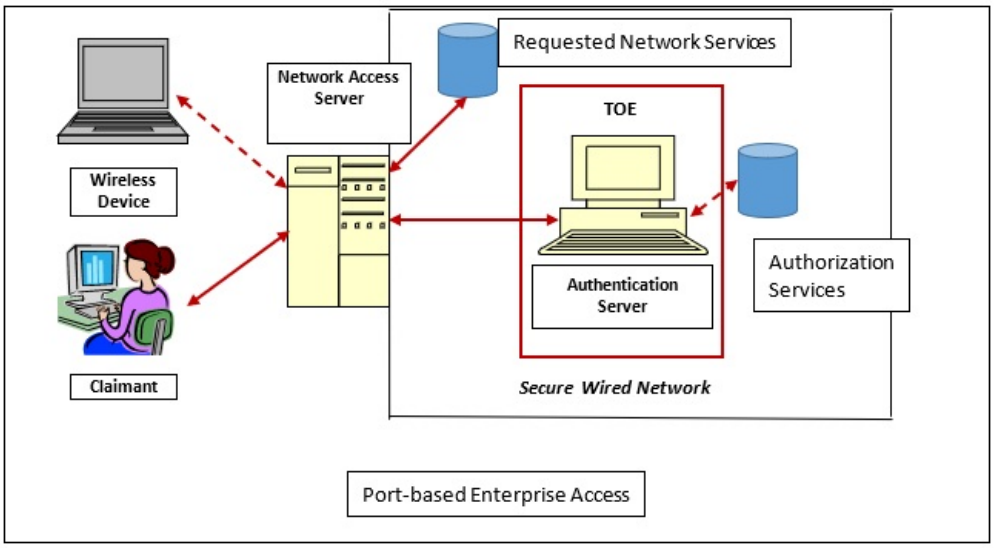


Figure 1: NAS with an Authentication Server

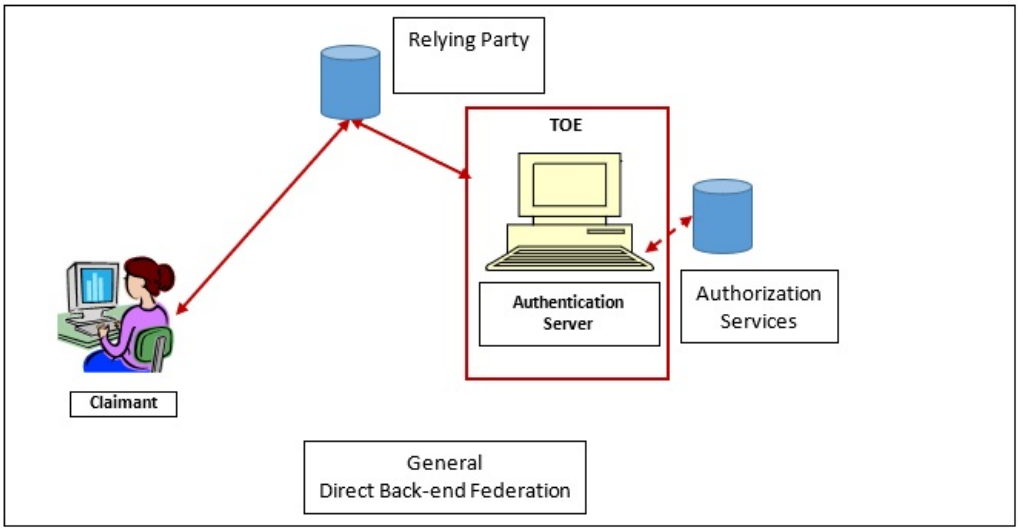


Figure 2: Generic Authentication Server User Case

1.5 Use Cases

This PP-Module defines the potential use cases below for the authentication server TOE. Use Case 1 defines the physical embodiment of the TOE, while Use Cases 2-4 define its role in a network infrastructure.

[USE CASE 1] Dedicated Appliance

The authentication server is integrated on a standalone network appliance. In this use case, conformance to the NDcPP and this PP-Module is sufficient to ensure security. This PP-Module does not cover the use case where the authentication server is an application that is installed on a general-purpose computer.

[USE CASE 2] Standalone Server

The system on which the authentication server is deployed is solely responsible for acting as an authenticator. In this deployment, the authentication server's only network infrastructure role is to communicate with the relying party for receiving challenges and issuing responses.

[USE CASE 3] Relying Party Co-Location

The system on which the authentication server is deployed acts as both the relying party or its proxy and the authentication server. In this deployment, the authentication server's interactions with the relying party are internal-only. Regardless of whether the relying party is a standalone component or the authentication server executable code also provides relying party functionality, the TOE's logical boundary still only includes the authentication server component. Additionally, if the authentication server is a software application that can be deployed independent of the relying party, the required external trusted communications must still be supported; an authentication server cannot use the fact that it can be deployed on the same physical server as the relying party as a way to exempt itself from implementation of IPsec, RadSec, or mutually authenticated (D)TLS with an external relying party.

[USE CASE 4] Integrated as an Authorization Server Component

The system on which the authentication server is deployed also acts as an authorization server (e.g., as part of an AAA server) that provides authorization services in addition to the authentication server. Assertions made via the direct connection can also include authorization information, and an unauthorized, but authenticated user may result in a negative response to the relying party. Regardless of whether these are all standalone components or the authentication server executable code also provides authorization functionality, the TOE's logical boundary still only includes the authentication server component. As in the case where the authentication server is co-located with the relying party, this deployment does not exempt the TOE from being able to implement all the functionality that this PP-Module requires.

2 Conformance Claims

Conformance Statement

This PP-Module inherits exact conformance as required from the specified Base-PP 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 may be specified in a PP-Configuration with this PP-Module other than the Base-PP specified in Section 1.1 Overview.

CC Conformance Claims

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

Package Claims

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.

3.1 Threats

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

T.FALSE_ENDPOINTS
A malicious actor may falsely impersonate the TOE or a federated relying party in order to cause the TOE to operate in an insecure manner or to extract security-relevant, or sensitive user data from the TOE or its Operational Environment.

T.INVALID_USERS
A malicious user may supply incorrect or insufficient credential data or an otherwise invalid authentication request that is approved or ignored by the TSF such that protected resources are subject to unauthenticated access.

3.2 Assumptions

These assumptions are made on the Operational Environment (OE) in order to be able to ensure that the security functionality specified in the PP-Module can be provided by the TOE. If the TOE is placed in an OE that does not meet these assumptions, the TOE may no longer be able to provide all of its security functionality. All assumptions for the OE of the Base-PP also apply to this PP-Module.

A.RP_FEDERATION
It is assumed that the TOE is federated with one or more relying parties that transmit authentication requests to it.

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.AUTH_POLICY
The organization defines, for each protected resource, an authentication policy that specifies the authenticators that must be provided to access a given resource.

4 Security Objectives

4.1 Security Objectives for the TOE

O.AUTHORIZED_USE
The TOE shall provide mechanisms that prevent and detect its unauthorized use.

O.SECURITY_ASSOCIATION
The TOE shall provide the information to the relying party to enable it to verify that the claimant has possession of an authentication key.

O.TRUSTED_RP
The TOE shall provide mechanisms to authenticate itself to a federated RP and authenticate a federated RP before providing an identity assertion.

O.USER_AUTH
The TOE shall provide a mechanism to assess authentication requests and respond with an authentication assertion based on data that is supplied in the request.

4.2 Security Objectives for the Operational Environment

All objectives for the OE of the Base-PP also apply to this PP-Module.

OE.RP_FEDERATION
The TOE will be deployed in such a manner that it is federated with one or more relying parties that transmit authentication requests to it.

OE.REQUIRE_AUTH
The operational environment will protect assets in a manner that requires authentication commensurate with the sensitivity of the assets.

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.FALSE_ENDPOINTS	O.TRUSTED_RP	The TOE's enforcement of mutual authentication allows it and the relying party to identify and reject attempts for each component to be impersonated.
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS (from NDcPP)	O.AUTHORIZED_USE	YYYYY
T.UNDETECTED_ACTIVITY (from NDcPP)	O.AUTHORIZED_USE	YYYYY
T.INVALID_USERS	O.SECURITY_ASSOCIATION	The TOE's ability to maintain a security association ensures that a mechanism exists for the TSF to assert to an external entity that a given user is valid.
	O.USER_AUTH	The TOE's proper implementation of the claimant authentication ensures that it will accurately process authentication attempts to allow only valid authentication attempts. The TOE's ability to use trusted communications as part of the federation protocol implementation ensures that modification or disclosure of authentication data cannot be used as a method to gain access to credentials or modify an authentication result.
A.RP_FEDERATION	OE.RP_FEDERATION	The OE objective OE.RP_FEDERATION is realized through A.RP_FEDERATION.
P.AUTH_POLICY	OE.REQUIRE_AUTH	Definition of an authentication policy, which can be enforced through deployment of a conformant TOE, can be used to ensure that organizational assets are protected by enforcing appropriate

requirements for authentication.

5 Security Requirements

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

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

5.1 NDcPP Security Functional Requirements Direction

In a PP-Configuration that includes the NDcPP, the TOE is expected to rely on some of the security functions implemented by the Authentication Server as a whole and evaluated against the NDcPP. The following sections describe any modifications that the ST author must make to the SFRs defined in the NDcPP in addition to what is mandated by Section 5.2 TOE Security Functional Requirements.

5.1.1 Modified SFRs

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

5.1.1.1 Identification and Authentication (FIA)

FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev

This SFR is selection-based in the NDcPP. When the TOE conforms to this PP-Module it is mandatory because of the PP-Module's certificate-based authentication of the channel between the TOE and a federated relying party.

Evaluation Activities

FIA_X509_EXT.1/Rev
TSS
There are no additional TSS evaluation activities for this component beyond what the NDcPP requires.

Guidance
There are no additional guidance evaluation activities for this component beyond what the NDcPP requires.

Tests
There are no additional test evaluation activities for this component beyond what the NDcPP requires.

FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1

This SFR is selection-based in the NDcPP. When the TOE conforms to this PP-Module it is mandatory because of the PP-Module's requirement for certificate-based authentication of the channel between the TOE and a federated relying party using IPsec with certificate-based authentication, mutually authenticated TLS or DTLS, or RADsec, which in turn uses IPsec or mutually authenticated TLS or DTLS.

Evaluation Activities

FIA_X509_EXT.2
TSS
There are no additional TSS evaluation activities for this component beyond what the NDcPP requires.

Guidance
There are no additional guidance evaluation activities for this component beyond what the NDcPP requires.

Tests
There are no additional test evaluation activities for this component beyond what the NDcPP requires.

FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1

This SFR is selection-based in the NDcPP. When the TOE conforms to this PP-Module it is mandatory because of the PP-Module's requirement for implementation of mutually-authenticated TLS or DTLS.

Evaluation Activities

FIA_X509_EXT.3
TSS
There are no additional TSS evaluation activities for this component beyond what the NDcPP requires.

Guidance
There are no additional guidance evaluation activities for this component beyond what the NDcPP requires.

Tests
There are no additional test evaluation activities for this component beyond what the NDcPP requires.

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 Security Audit (FAU)

FAU_GEN.1/AuthSvr Audit Data Generation (Authentication Server)

FAU_GEN.1.1/AuthSvr

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

a. Start-up and shutdown of the audit functions;
b. All auditable events for the [*not specified*] level of audit; and
c. [*Auditable events listed in the Auditable Events table (Table 2)*

Requirement	Auditable Events	Additional Audit Record Contents
FCO_NRO.1	Claimant request for which the TOE does not have credential verification data	Identity of the claimant
FCO_NRR.1	None	
FCS_CKM.3	[*selection: attempt to export plaintext key or CSP via defined interface, none*] Note: if no defined interfaces have access to persistent keys or CSP, select 'none'	If attempt is detected, record process identifier, authorized user's identifier (if any)

FCS_EAPTLS_EXT.1	Protocol failures	If failure occurs, record a descriptive reason for the failure
	Successful and failed authentication of claimant	Identifier of claimant
FCS_RADIUS_EXT.1	Protocol failures	If failure occurs, record a descriptive reason for the failure
	Success/failure of authentication	None
FCS_RADSEC_EXT.1 (selection-based)	None	
FCS_STG_EXT.1	None	
FIA_AFL.1/AuthSvr	The reaching of the threshold for the unsuccessful authentication attempts	The claimed identity of the entity attempting to authenticate or the IP where the attempts originated
	Disabling an account due to the threshold being reached	
FIA_HOTP_EXT.1 (selection-based)	Generation of a HOTP seed key	Entity identifier
	Entity HOTP value comparison	Result of comparison - success or failure
FIA_PSK_EXT.1/AuthSvr (selection-based)	None	
FIA_PSK_EXT.2 (selection-based)	None	
FIA_PSK_EXT.3 (selection-based)	None	
FIA_TOTP_EXT.1 (selection-based)	Generation of a TOTP seed key	Entity identifier
	Entity TOTP value comparison	Result of comparison - success or failure
FIA_X509_EXT.1/AuthSvr	Certificate validation failure	Reason for failure
FIA_UAU.6	All use of the authentication mechanism	Origin of the attempt (e.g., IP address)
FMT_SMF.1/AuthSvr	All management actions	Identifier of initiator
FTA_TSE.1	Denial of session establishment due to the session establishment mechanism	Reason for denial, origin of establishment attempt
FTP_ITC.1/NAS	Initiation of the trusted channel	Identification of the initiator
	Termination of the trusted channel	Identification of the initiator
	Failure of the trusted channel functions	Target of failed trusted channels establishment attempt

Table 2: Auditable Events

Application Note: The auditable events defined in Table 2 are for the SFRs that are explicitly defined in this PP-Module and are intended to extend FAU_GEN.1 in the Base-PP.
The events in the Auditable Events table should be combined with those of the NDcPP in the context of a conforming Security Target.

The Auditable Events table (Table 2) includes selection-based SFRs. The auditing of selection-based SFRs is only required if that SFR is included in the ST.

Per FAU_STG_EXT.1 in the Base-PP, the TOE must support transfer of the audit data to an external IT entity using a trusted channel.

FAU_GEN.1.2/AuthSvr

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

a. Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b. For each audit event type, based on the auditable event definitions of the functional components included in the PP-**Module**/ST, [*information specified in column three of the Auditable Events table (Table 2)*].

Evaluation Activities

FAU_GEN.1/AuthSvr
TSS
There are no TSS evaluation activities for this SFR.

Guidance
The evaluator shall ensure that the operational guidance identifies the auditable events and includes representative examples of each event so that the presentation of each event can be identified.

Tests
- *Test 1:*
  The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records in accordance with the evaluation activities associated with the functional requirements in this PP-Module. When verifying the test results, the evaluator will ensure the audit records generated during testing match the format specified in the administrative guide and that the fields in each audit record have the proper entries.

  Note that the testing here can be accomplished in conjunction with the testing of the security mechanisms directly.

5.2.2 Communications (FCO)

FCO_NRO.1 Selective Proof of Origin

FCO_NRO.1.1

The TSF shall be able to generate evidence of origin for transmitted [*identity authentication assertions, [**selection**: authentication requests, IKE authentication phase security associations, [**assignment**: claimant identity attributes], no other data]*] at the request of the [*relying party, [**selection**: external authentication servers in support of pass-through, no other entities]*].

Application Note: The intent of this requirement is for the TOE to provide the source of origin (non-repudiation) for assertions and sensitive data associated to claimants it provides to relying parties. The ST author will claim 'authentication requests' and 'external authentication servers…' if the TSF supports pass-through communication with external authentication servers. The ST author claims additional information provided to a relying party via an authenticated channel as appropriate.

FCO_NRO.1.2

The TSF shall be able to relate the [*authenticator*] of the originator of the information, and the [*authentication request*] of the information to which the evidence applies.

Application Note: The intent of this requirement is for the TOE to be able to associate authentication assertions it makes to requests made to it by a relying party. For pass-through functionality, the TOE relates requests and response messages it forwards between external entities via identity information asserted in the EAP headers.

FCO_NRO.1.3

The TSF shall provide a capability to verify the evidence of origin of information to [*recipient*] given [*an authenticated channel is established with a trusted relying party]*].

Evaluation Activities

FCO_NRO.1

TSS

The evaluator shall ensure that the ST includes a description of authentication assertions, security associations or sensitive data associated with a claimant that is provided to a relying party, and a description of each protocol that carries such data.

If pass-through is supported, the evaluator shall ensure that the ST includes a description of the claimant authentication methods allowed and the method used to mutually authenticate to external authentication servers.

The evaluator shall verify that the descriptions indicate how the TSF authenticates itself to the external entities via those protocols, and that no data is passed via an unauthenticated protocol.

The evaluator shall verify that the ST describes how the TSF handles session interruptions and resumptions to ensure the relying party is able to associate data associated with a claimant to the authentication request by the relying party and the authenticator provided by the claimant.

Guidance

The evaluator shall ensure that any instructions for configuring the TSF to meet the requirements are provided.

Tests

The evaluator shall perform the following tests:

- *Test 2:*
  - *Step 1: The evaluator shall establish a connection with the TSF from two trusted relying parties RP1 and RP2 and verify that each of RP1 and RP2 are able to authenticate the TOE.*
  - *Step 2: The evaluator shall initiate an authentication request for a claimant C1 via RP1, providing valid authentication data, and verify that RP1 receives an authentication assertion via the authenticated channel indicating C1 is authenticated.*
  - *Step 3: The evaluator shall initiate an authentication request for a claimant C2 via RP2, providing invalid authentication data and confirm that the TOE does not provide an authentication assertion indicating C2 is authenticated via the authenticated channel.*
  - *Step 4: The evaluator shall send correct authentication data associated with claimant C2 via RP1 without sending a new authentication request and observe that the TOE ignores the request.*
- *Test 3: [conditional on support for pass-through] The intent of this test is to demonstrate the TSF is able to authenticate to external entities for registered users over a pass-through method, and ignores requests for non-registered users.*
  - *Step 1: The evaluator shall follow operational guidance to configure the TOE to connect to an external authentication server using pass-through functionality, and initiate a request from a trusted relying party that results in the TSF exercising pass-through functionality to authenticate a registered claimant.*
  - *Step 2: The evaluator shall observe that the TSF authenticates to the external authentication server prior to sending any authentication requests.*
  - *Step 3: The evaluator shall then follow operational guidance to de-register the claimant at the TOE, and ensure the claimant is still registered at the external authentication server. The evaluator shall repeat initiation of the authentication request for the claimant, and observe that the TSF associates the identifier of the request by dropping the request without forwarding.*

FCO_NRR.1 Selective Proof of Receipt

FCO_NRR.1.1

The TSF shall be able to generate evidence of receipt for received [*authentication requests, [**selection**: authentication responses and queries, none]*] at the request of the [*originator*].

Application Note: The intent of this requirement is for the TOE to be able to return a valid response to the relying party upon receipt of an Access-Request. If the TSF supports pass-through functionality, the ST author claims ‘authentication responses and queries’ in the selection for authentication in communications with external authentication servers.

FCO_NRR.1.2

The TSF shall be able to relate the [*claimant identifier and claimant authenticators*] of the recipient of the information, and the [*identity assertion, information requests, and error responses*] of the information to which the evidence applies.

Application Note: The intent of this requirement is for the ST author to list the information supplied by the TOE in response to an authentication request that confirms receipt of the request, and identifies:

- the authentication request that is being responded to;
- the mutually authenticated channel between the trusted relying party and the TOE.

FCO_NRR.1.3

The TSF shall provide a capability to verify the evidence of receipt of information to [*originator*] given [*establishment of a mutually authenticated channel with a trusted relying party*].

Evaluation Activities

FCO_NRR.1
TSS
The evaluator shall ensure that the ST includes a description of each messaging protocol and the specific messages provided to a relying party in response to authentication requests, to include any affirmative and negative responses, and requests for additional information.

The evaluator shall verify that the descriptions specify how the TSF indicates the identity of the claimant associated with any responses to a request.

The evaluator shall verify that the ST describes how the TSF handles session interruptions and resumptions to ensure the relying party is able to associate data associated with a claimant to the authentication request by the relying party and the authenticator provided by the claimant.

Guidance
The evaluator shall ensure that any instructions for configuring the TSF to meet the requirements are provided.

Tests
For each messaging protocol supported, the evaluator shall perform the following test:

- *Test 4: The evaluator shall establish a connection between a trusted relying party and the TOE and send an authentication request for a registered claimant, in accordance with the messaging protocol standard. The evaluator shall confirm the TOE responds to each message sent by the relying party with a message that appropriately identifies the claimant and confirms receipt of the request.*

5.2.3 Cryptographic Support (FCS)

FCS_CKM.3 Cryptographic Key Access

FCS_CKM.3.1

The TSF shall perform [*access control for persistent private and secret keys and critical security parameters required by this PP-Module*] in accordance with a specified cryptographic key access method [*ensuring only authorized security functionality can access plaintext keys or critical security parameters*] that meets the following: [*keys and critical security parameters are not exportable in plaintext and keys and critical security parameters are not viewable in plaintext*].

Application Note: Exposure of keys used for assertion signatures, including private keys associated to certificates used to establish a protected channel to relying parties and claimants, one-time-password seed keys, and plaintext passwords can undermine or bypass the protections required for TOE functionality. The ST author describes the specific methods used to prevent unauthorized or unnecessary access to these keys and critical security parameters. This requirement is not intended to cover unanticipated exploits; it is only required that plaintext keys and critical security parameters not be exportable or viewable by defined interfaces. OTP seed key values are shared using out-of-band methods with the associated entities. This requirement implies that the method to export these values uses encrypted key transport methods.

Evaluation Activities

FCS_CKM.3
TSS
The evaluator shall verify the ST includes a description of all persistent secret and private keys used by the TSF to perform functions in this PP-Module. The evaluator shall verify the ST describes mechanisms used to prevent unauthorized exposure of keys.

Guidance
The evaluator shall verify that any configuration required to meet the requirements are described.

Tests
The intent of these tests is to ensure keys are not accessible using common interfaces and functionality of the TSF. It is not intended for the evaluator to attempt to cause a system crash in order to read keys and critical security parameters directly from memory or to modify functionality of the TSF.

The evaluator shall perform the following tests:

- *Test 5: The evaluator shall attempt to export each key and critical security parameter using available interfaces and verify the mechanism is effective at preventing exposure of the key in plaintext.*
- *Test 6: The evaluator shall assume each of the privileged user roles and attempt to gain read access to each of the keys and critical security parameters via available interfaces.*

FCS_EAPTLS_EXT.1 EAP-TLS Protocol

FCS_EAPTLS_EXT.1.1

The TSF shall implement [**selection**: *EAP-TLS as specified in RFC 5216*, *EAP-TTLS as specified in RFC 5881*] as updated by RFC 8996 with [**selection**: *TLS*, *DTLS*] implemented using mutual authentication in accordance with [**selection**: *FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2*, *FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2*].

FCS_EAPTLS_EXT.1.2

The TSF shall generate random values used in the [**selection**: *EAP-TLS*, *EAP-TTLS*] exchange using the RBG specified in FCS_RBG_EXT.1.

FCS_EAPTLS_EXT.1.3

The TSF shall support claimant authentication using certificates and [**selection**: *static PSK*, *HOTP*, *TOTP*, *other authentication-verification methods via pass-through*, *no other methods*].

FCS_EAPTLS_EXT.1.4

The TSF shall not forward an EAP-Success response to the relying party if the client certificate is not valid according to FIA_X509_EXT.1**/AuthSvr**, if the [**selection**: *TLS*, *DTLS*] session is not established, or if any of [**selection**: *PSK*, *HOTP value*, *TOTP value*, *no other authenticator*] required by the authentication policy are not provided or if any of the required authenticators presented in the authentication request is not valid.

Application Note: The ST author should indicate support for EAP-TLS or EAP-TTLS in FCS_EAPTLS_EXT.1.1. In the third selection, 'FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2' or 'FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2' is selected according to the TLS or DTLS support indicated in the second selection, with the expectation that the corresponding SFRs from the Base-PP are claimed.

The selection in FCS_EAPTLS_EXT.1.2 matches the first selection in FCS_EAPTLS_EXT.1.1.

The selections made in FCS_EAPTLS_EXT.1.3 may trigger the inclusion of selection-based SFRs, as follows:

- Any of "static PSK," "HOTP," or "TOTP" requires inclusion of FIA_PSK_EXT.1/AuthSvr.
- "HOTP" requires inclusion of FIA_HOTP_EXT.1.
- "TOTP" requires inclusion of FIA_TOTP_EXT.1

The ST author claims any additional supported authentication-verification methods in FCS_EAPTLS_EXT.1.3. Each supported method is claimed independently, even if combinations of the methods are required for individual claimant authentication. For any authentication methods that are only supported by pass-through functionality, the ST author should claim 'other authentication-verification methods via pass-through' without claiming the corresponding method in the same selection. Pass-through functionality can typically support any authentication method, including ones not specified in the SFR. However, it

is preferred that the TSF not use pass-through functionality for EAP methods that do not align to standardized methods utilizing certificate-based authentication of the claimant.

Evaluation Activities

FCS_EAPTLS_EXT.1

TSS

The evaluator shall examine the ST to ensure the EAP protocol is described in accordance with the claimed RFC. For each supported mode, the evaluator shall ensure the ST describes the following:

- *The mechanism to authenticate a claimant uses (D)TLS with client certificate authentication in combination with any other supported authenticator outputs, and any configurable features.*
- *The source of randomness meets FCS_RBG_EXT.1 for use in key and nonce generation for the underlying (D)TLS channel and supported authentication methods.*

The evaluator shall also verify that the ST contains a description of the user access policy, including which authenticator outputs are required under the default configuration and which features of the user access policy are configurable.

Guidance

The evaluator shall ensure that the operational guidance includes any instructions for configuring the TOE to support the claimed functions.

If any features of the access control policy are configurable (e.g., the supported authentication mechanism), the evaluator shall confirm that the operational guidance describes how to configure these features.

Tests

The evaluator shall perform the following tests:

- *Test 7: TLS/DTLS testing is performed as part of FCS_TLSS_EXT.1 and .2 or FCS_DTLSS_EXT.1 and .2. When TLS/DTLS cannot be invoked directly using available TOE interfaces, the test procedures are modified in the following manner:*
  - *When required to send a client handshake to the TOE, the evaluator shall establish a connection with the test relying party and send the specified TLS client handshake messages in response to requests from the test relying party.*
  - *The evaluator ensures the test relying party encapsulates the TLS handshake messages received from the test client and forwards the EAP messages to the TOE. Alternatively the evaluator may use a test relying party to modify client handshake messages as specified. When required to observe TLS server responses produced by the TOE, the evaluator shall ensure the test relying party properly extracts the TLS messages from the EAP messages, and shall observe the response received at the test TLS client to verify that the TOE responds as indicated in test procedures. Alternatively, the evaluator may extract and reconstruct TLS responses received within the EAP messages received from the TOE at the test relying party.*
- *Test 8: EAP testing: For each EAP mode supported, the evaluator shall perform any configuration of the TOE necessary to select the desired mode according to the operational guidance and perform the following tests:*
  - *Test 8.1: The evaluator shall determine the user access policy enforced by the TOE (if the TOE has a configurable user access policy, the evaluator may configure the TOE according to operational guidance to require claimant authentication using only a certificate to simplify subsequent test procedures). The evaluator shall initiate an authentication request from a test relying party to the TOE for a valid claimant registered with the TOE. The evaluator shall observe that once the EAP identity is established, the TOE sends an EAP request indicating the expected EAP mode (EAP-TLS or EAP-TTLS) and having the start-bit set.*

    The evaluator shall ensure a TLS client hello message from the valid claimant at a test client is EAP-encapsulated by the test relying party and provided to the TOE. The evaluator shall observe that the TOE responds with an EAP-encapsulated hello message to include a certificate request message.

    The evaluator shall ensure the test client successfully completes the TLS handshake, and the test relying party properly encapsulates the TLS messages, to include the client finished message, and observes that the TOE responds in a manner indicating the TLS channel was successfully established. Note – if the user access policy is to only require certificate verification, then the expected response is an EAP-Success message. If the user access policy requires additional factors to be supported under EAP-TTLS, additional EAP-TTLS messages must be sent to the test relying party to request those additional factors from the test client. These are encrypted under the TLS tunnel established between the claimant and the TOE. In this case, the evaluator observes these requests at the test client to confirm the certificate verification was successful.

- *Test 8.2: The evaluator shall initiate an authentication request from the test relying party for a valid claimant registered with the TOE, different than the claimant used for Test 8.1. The evaluator shall send appropriate encapsulated TLS handshake messages to the TOE, to include a valid certificate response, but send an EAP-encapsulation of a modified client finished message to the TOE. The evaluator shall observe that the TOE does not send an EAP-Success message; the TOE is allowed either to send an EAP-request message to initiate a new TLS handshake or an EAP-Failure message.*
- *Test 8.3:*

  [conditional on support for additional authentication factors under EAP-TTLS]: For each combination of authentication factors supported by the TOE's user authentication policy, the evaluator shall follow the operational guidance to configure the TOE's user access policy to require the desired combination. The evaluator shall initiate an authentication request from a test client with a registered claimant having valid credentials for all factors. The evaluator shall observe that the TOE responds to the authentication request with an exchange of EAP-requests to successfully establish a mutually authenticated TLS/DTLS tunnel and that on completion, the TOE provides additional EAP-requests that when decrypted at the test client, results in prompts for additional factors.

  For each additional factor in the combination, the evaluator shall input an incorrect value for requested authentication factors and observe that the TOE responds with an EAP-request that prompts the claimant to re-enter the value. The evaluator shall then input the correct value and observe that the TOE responds with an EAP-request resulting in a prompt for the next factor. The evaluator shall continue, in turn entering first, invalid, and then valid entries until all factors have been successfully provided. The evaluator shall confirm that on successful submission of valid factors, the TOE sends an EAP-Success message to the test relying party.

  If combinations are supported, the TSF may request each factor individually, or request all factors in a particular order in a single request. If multiple factors are included in a request, the evaluator shall test each component, observing that the TSF will reject the entry until all components are correct.

FCS_RADIUS_EXT.1 Authentication Protocol

FCS_RADIUS_EXT.1.1

The TSF shall implement the [**selection**: *RADIUS protocol as specified in RFC 2865*, *DIAMETER protocol as specified in RFC 6733*, [***assignment**: other direct identity federation protocol]*] for communication of identity and authentication information with a relying party.

FCS_RADIUS_EXT.1.2

The TSF shall implement encapsulated EAP in accordance with FCS_EAPTLS_EXT.1.

FCS_RADIUS_EXT.1.3

The TSF shall provide [**selection**: *a key indicator*, *an encrypted parameter*, *an encrypted value*] for a key held by the successfully authenticated claimant derived from the supported EAP mode and provided to the relying party in accordance with the protocol indicated in FCS_RADIUS_EXT.1.1.

Application Note: The ST author describes how the TSF communicates with a relying party at the application layer to receive authentication requests and provide identity assertions. RADIUS and DIAMETER protocols are used with AAA servers when the relying party is a NAS. However, other direct access identity federation protocols that support FCS_EAPTLS_EXT.1 and identify a key held by the authenticated claimant that can be confirmed by the relying party are acceptable. If other protocols are claimed, the ST author includes the RFCs and indicates the messages used for authentication requests and assertions.

The ST author indicates which keys held by the authenticated claimant are available to the relying party for key-holder verification. For RADIUS and DIAMETER, the EAP-TLS/EAP-TTLS master key established during the TLS handshake with the claimant is shared with the relying party, encrypted under the protected channel between the TSF and the relying party. Both the relying party and claimant derive the AUTH MSK/security association for an IPsec session from this master key. More generally, a key indicator can be a reference identifier for a shared secret key, or a public key, certificate, or other identifier associated with a private asymmetric key controlled by the authenticated claimant.

Evaluation Activities

FCS_RADIUS_EXT.1
TSS
The evaluator shall review the ST to ensure the supported protocols are described and that the description includes the following:

- *Types of claimant-held keys that can be used by the relying party for key-holder verification in accordance with the supported EAP mode claimed in FCS_EAPTLS_EXT.1.*
- *How information provided by the TOE to the relying party allows the relying party to perform key-holder verification using the key.*
- *How key related information provided by the TOE is protected in transit to the relying party.*

Guidance
The evaluator shall verify that all configurable features of the TSF are described, and that instructions are provided to meet the requirements.

Tests
The evaluator shall perform the following test in conjunction with testing for FCS_EAP-TLS_EXT.1 after successful authentication:

- *Test 9: For each type of claimant held key supported, the evaluator shall confirm that communication between the test client and the test relying party encrypted using the indicated key is successful.*

FCS_STG_EXT.1 Cryptographic Key Storage

FCS_STG_EXT.1.1

Persistent private and secret keys shall be stored within the TSF [**selection**:

- *encrypted within a hardware protected key*
- *in a hardware cryptographic module*
- *within an isolated execution environment protected by a hardware key*

].

Evaluation Activities

FCS_STG_EXT.1
TSS
The evaluator shall verify the TSS includes a description of protected key storage.

Guidance
The evaluator shall verify that the operational guidance includes any information needed to configure the TOE to meet this requirement.

Tests
There are no test EAs for this component.

5.2.4 Identification and Authentication (FIA)

FIA_AFL.1/AuthSvr Authentication Failure Handling (Claimant)

FIA_AFL.1.1/AuthSvr

The TSF shall detect when [*an administrator configurable positive integer **of successive***] unsuccessful authentication attempts occur related to [*claimants attempting to authenticate*].

FIA_AFL.1.2/AuthSvr

When the defined number of unsuccessful authentication attempts has been [*met*], the TSF shall [**selection, choose one of**: *prevent the offending remote entity from successfully authenticating until [**assignment**: action] is taken by a local Administrator, prevent the offending claimant from successfully authenticating until an administrator-defined time period has elapsed*].

Application Note: This requirement applies to claimant authentication attempts in support of an authentication service provided for a federated relying party. This requirement does not apply to login to the TOE by privileged users for administrative accesses; these cases are addressed by the Base-PP iteration of this SFR. Responses to authentication queries to aid the claimant in providing acceptable authenticators is not considered a preventative action and are allowed prior to reaching the lockout threshold. The “action” taken by a local administrator is implementation specific and is defined in the operational

guidance (for example, lockout reset or password reset). The ST author chooses one of the selections for handling of authentication failures depending on how the TOE has implemented this handler.

Evaluation Activities

FIA_AFL.1/AuthSvr
TSS
The evaluator shall examine the TSS to verify that it contains a description of how successive unsuccessful authentication attempts by claimants are detected and tracked. The evaluator shall verify that the TSS describes the method by which the offending claimant is prevented from successfully being authenticated by the TOE, and the actions necessary to restore this ability.

Guidance
The evaluator shall examine the operational guidance to verify that it describes how to configure the threshold for unsuccessful claimant authentication attempts, what the acceptable range of values for that threshold is, and how to perform any actions that affect claimants that are limited in this manner (e.g., instructions for configuring the lockout period or for manually unlocking the offending claimant).

Tests
The evaluator shall perform the following tests in conjunction with testing for FCS_EAPTLS_EXT.1 Test 8.1, and if applicable, Test 8.3 for each claimant authentication method:

- *Test 10: The evaluator shall follow the operational guidance to configure a number of failed attempts that will cause lockout behavior to be enforced against a claimant. The evaluator shall establish a registered user and provide invalid input for the authentication method repeatedly to reach the configured limit. The evaluator shall then observe the configured penalty is imposed.*
- *Test 11: If the administrator action selection is claimed in FIA_AFL.1.2/AuthSvr, the evaluator shall ensure that following the operational guidance for restoring access to a locked-out claimant will subsequently allow that claimant to be authenticated.*

  If the time period selection is claimed in FIA_AFL.1.2/AuthSvr, the evaluator shall follow the operational guidance to configure a certain lockout time for claimants that are locked out due to excessive authentication failures. The evaluator shall cause a claimant to be locked out in this manner, wait for a time period that is just less than the configured value, and verify that an authentication attempt using valid credentials still does not result in successful access. The evaluator shall then repeat this behavior but wait for just after the configured time period has elapsed to show that an authentication attempt using valid credentials results in successful access.

FIA_UAU.6 Re-Authenticating

FIA_UAU.6.1

The TSF shall re-authenticate the **administrative** user under the conditions [*when the user changes their password, [**selection**: following TSF-initiated session locking, [**assignment**: other conditions], no other conditions]*].

Evaluation Activities

FIA_UAU.6
TSS
There are no TSS evaluation activities for this component.

Guidance
The evaluator shall ensure that the operational guidance includes instructions on how an administrator of the TOE can change their own password.

Tests

- *Test 12: The evaluator will access the TOE using a particular administrative account and then attempt to change the password of that account as directed by the operational guidance. While making this attempt, the evaluator will verify that re-authentication is required.*

  If other re-authentication conditions are specified, the evaluator shall cause those conditions to occur and verify that the TSF re-authenticates the authenticated user.

FIA_X509_EXT.1/AuthSvr X.509 Certificate Validation (Claimant)

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

- RFC 5280 **version 3** certificate validation and certification path validation **supporting [selection:** ***a minimum path length of [assignment: value greater than or equal to three], no prior constraints on path length*** **]**
- The certification path must terminate with a CA certificate **trusted by the TSF specifically for claimant authentication.**
- 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 **each certificate in the certificate path [selection:**
  - ***containing an OCSP provider in the AIA extension using the Online Certificate Status Protocol (OCSP) as specified in RFC 6960***
  - ***containing a certificate revocation list (CRL) distribution point in the CRLDP extension or AIA extension using [selection: a CRL as specified in RFC 5280 Section 6.3, a CRL as specified in RFC 5759 Section 5]***

  ].
- The TSF shall validate the extendedKeyUsage field **is present and contains key usage values** according to the following rules: **[selection:**
  - ***Certificates do not assert anyExtendedKeyUsage (OID 2.5.29.37.0)***
  - ***Client certificates associated with authenticated entities presented for [selection: TLS, DTLS] shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.***
  - ***[selection:***
    - ***Server certificates presented for [selection: TLS, DTLS] shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.***
    - ***Certificates presented for IPsec shall have the ipsec-IKE purpose (id-kp 17 with OID 1.3.6.1.5.5.7.3.17)***
    - ***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.***

    ]

  ].
- **The TSF shall validate that each CA certificate in the certification path indicating a path length constraint in the basicConstraints extension does not have more than the specified number of subordinate CA certificates in the certification path from the end-entity certificate to the CA certificate indicating the constraint, not counting the CA certificate itself or any self-issued certificates in the certification path.**
- **The TSF shall process name constraints of type Directory Name and [selection:** ***rfc822Name, dnsName, UPN Name (Other Name = id-ms-san-sc-logon-upn), [assignment: other name type], no other name type*** **] by verifying that each name of a supported name type present in the end-entity certificate subject field or subjectAlternateName extension, is allowed in each CA certificate in the certification path, is not disallowed by any of the CA certificates in the certification path, and that each name type included in the end-entity certificate and constrained by a CA certificate in the certification path is processed.**
- **The TSF shall process the following certificate extensions: [selection:**
  - ***Certificate Policy extension in accordance with RFC 5280 and [selection:***
    - ***Policy mapping extension in accordance with RFC 5280***
    - ***Policy constraints extension in accordance with RFC 5280***
    - ***Inhibit anyPolicy extension in accordance with RFC 5280***
    - ***No other policy related extension***

    ] in support of claimant authentication and [assignment: other intended purposes and limitations of policy related extension processing]
  - ***[assignment: other standard extensions]***

- ***no other extensions***

]**.**

Application Note: This SFR is iterated from the Base-PP to define X.509 validation requirements that are specific to claimaint authentication.

The ST author claims supported certificate validity checking options for each rule. For name constraints, all supported name types used to match names presented in a certificate to registered users and the associated standard matching method are described.

The ST author claims supported certificate policies. 'Policy Constraints…' is claimed if the TOE's authentication policy depends on the certificate policies for claimant certificates. Other policy related extensions within the selection are claimed if supported. The extension inhibitPolicyMapping is not claimed if the TSF does not support certificate chains of length four or more. The policy related extensions, if supported, are primarily used in this PP-Module for claimant authentication, but are allowed for other certificate authentications. The ST author specifies the intended use and any limits of support for these extensions or specifies 'no other policy related extension' in the assignment of this selection.

The ST author specifies any additional supported X.509 extensions, and the associated extension processing rules used to determine claimant identity attributes or conditions that can be used in the TOE's authentication policy.

FIA_X509_EXT.1.2/AuthSvr

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.

Evaluation Activities

FIA_X509_EXT.1/AuthSvr

TSS

The evaluator shall ensure the TSS describes where the check of validity of the certificates takes place. The evaluator ensures the TSS also provides a description of the certificate path validation algorithm.

Guidance

The evaluator shall ensure that instructions for any configurable features of the validation process are included. If the ST includes provisions for exception processing of certificate revocation status information, the evaluator shall ensure the operational guidance contains instructions on how the indicated options are configured.

If the TOE supports processing of the policy constraints extension and the TOE requires configuration to validate the policy of a claimant certificate, the evaluator shall verify that the operational guidance includes instructions for configuring this behavior.

Tests

The evaluator shall perform the following tests. The tests for the extendedKeyUsage rules, name constraints and policy constraints, if supported, are performed in conjunction with the uses that require those rules.

- *Test 13: The evaluator shall demonstrate that validating a certificate without a valid certification path results in the function failing, for each of the following reasons, in turn:*
  - *by establishing a certificate path in which one of the issuing certificates is not a CA certificate*
  - *by omitting the basicConstraints field in one of the issuing certificates*
  - *by setting the basicConstraints field in an issuing certificate to have CA=False*
  - *by omitting the CA signing bit of the key usage field in an issuing certificate*
  - *by setting the path length field of a valid CA field to a value strictly less than the certificate path*

  The evaluator shall then establish a valid certificate path consisting of valid CA certificates and demonstrate that the function succeeds. The evaluator shall then remove trust in one of the CA certificates and show that the function fails.
- *Test 14: The evaluator shall demonstrate that validating an expired certificate results in the function failing.*
- *Test 15: The evaluator shall test that the TOE can properly handle revoked certificates – conditional on the revocation method that is selected; if multiple methods are selected, then the test is repeated for each method. The evaluator tests revocation for each certificate in the trust chain which advertises certificate status information. The evaluator shall ensure that a valid certificate is used, and that the validation function succeeds. The evaluator shall then attempt the test with a certificate that will be revoked (for each method chosen in the selection) and verify that the validation function fails.*
- *Test 16: [conditional] If any OCSP option is selected, the evaluator shall present a delegated OCSP certificate that does not have the OCSP signing purpose and verify that*

validation of the OCSP response fails. If CRL is selected, the evaluator shall configure the CA to sign a CRL with a certificate that does not have the cRLsign key usage bit set and verify that validation of the CRL fails.

- *Test 17: [conditional] If the TOE supports EC certificates, then the evaluator shall establish a valid, trusted certificate chain consisting of an EC leaf certificate, an EC Intermediate CA certificate not designated as a trust anchor, and an EC certificate designated as a trusted anchor, where the elliptic curve parameters are specified as a named curve. The evaluator shall confirm that the TOE validates the certificate chain.*
- *Test 18: [conditional] If the TOE supports EC certificates, then the evaluator shall replace the intermediate certificate in the certificate chain for Test 17 with a modified certificate. This certificate is modified to have the public key information field where the EC parameters use an explicit format version of the Elliptic Curve parameters in the corresponding field of intermediate CA certificate from Test 17. This certificate is also modified to be signed by the trusted EC root CA, but with no other changes. The evaluator shall confirm the TOE treats the certificate as invalid.*
- *Test 19: The evaluator shall test the following name constraints:*
  - *Test 19.1: For each name type supported, the evaluator shall establish a valid certificate for a registered entity. The evaluator shall ensure the certificate has a valid path length of at least three, consisting of a trusted root, an issuing CA that is not a trust anchor, and the leaf certificate representing the entity. The evaluator shall ensure that the leaf certificate includes a single name of the supported name type and no other DN or SAN entries. The evaluator shall initiate an application requiring authentication of that entity using the certificate and verify the TSF successfully authenticates the entity.*
  - *Test 19.2: For each leaf certificate used in Test 19.1, the evaluator shall establish a new leaf certificate that includes the same name and name type, but which is issued by a different subordinate CA asserting an allowed-list that does not include the name for the name-type. The evaluator shall ensure the subordinate CA is included in a valid chain to the same trusted root. The evaluator shall initiate the same application attempt as in Test 19.1 for the new certificate and observe that the TSF indicates the certificate is invalid.*
  - *Test 19.3: For each leaf certificate used in Test 19.1, the evaluator shall establish a new leaf certificate that includes the same name and name type, but which is issued by a different subordinate CA asserting a denylist matching the name for the name type. The evaluator shall ensure the subordinate CA is included in a valid certificate path to the same trusted root. The evaluator shall initiate the same application attempt as in Test 19.1 using the new certificate and observe that the TSF indicates the certificate is invalid.*
- *Test 20: [conditional] If the TOE supports processing of the policy constraints extension, then for each distinct purpose and within the constraints indicated in the ST (claimant authentication and any other supported subject types), the evaluator shall follow the operational guidance as necessary to configure the TOE to require the subject's certificate to assert a specific certificate policy. The evaluator shall perform the following sub-tests:*
  - *Test 20.1: The evaluator shall establish a certificate for the subject asserting the certificate policy OID required, issued by a Certification Authority also specifying the required certificate policy. The evaluator shall present the established certificate for authentication and verify that the TSF successfully validates the certificate.*
  - *Test 20.2: [conditional] If the ST selects 'Inhibit anyPolicy extension...," the evaluator shall repeat Test 20.1 using a certificate asserting the required policy but issued by a Certification Authority only asserting the 'anyPolicy' OID (value {2 5 29 32 0}) in its policy constraints extension. The evaluator shall observe that the TSF successfully validates the certificate.*
  - *Test 20.3: [conditional] If the ST indicates support for the policy mapping extension, the evaluator shall repeat Test 20.1 using a certificate asserting a new policy OID that does not match the required policy OID. This certificate is issued by a CA asserting the new policy OID in the policy constraints extension, and the CA certificate is issued by a second CA. This second CA asserts the required OID in its certificate constraints extension and contains a policy mapping extension including the mapping of the asserted policy to the required policy. The evaluator shall observe that the TSF successfully validates the certificate.*

    Note that installing a root CA trusted by the TOE with the required policy constraints and policy mapping extensions may be required if the TSF limits the path length of certificate chains.
  - *Test 20.4: [conditional] If the ST indicates support for both policy mapping and policy constraints extensions and also supports certificate chains of length four or more, the evaluator shall establish a certificate for the subject asserting a new policy OID that does not match the required policy OID. This certificate is issued by a CA asserting the new policy OID, which in turn is issued by a second CA which includes the policy mapping extension that maps the required policy OID to the new policy OID. The second CA is in turn issued by a third CA that has the extension policy constraints with the inhibitPolicyMapping field having value 0. The evaluator shall present the certificate to the TSF for authentication and observe that the TSF indicates the certificate is invalid.*

- *Test 20.5: [conditional] If the ST indicates support for both policy mapping and policy constraints extensions, the evaluator shall select a policy OID not required for authentication in the TOE's current configuration. The evaluator shall establish a certificate for the subject that does not assert the non-required policy, which is issued by a CA also asserting the new policy OID, which in turn, is issued by a CA asserting the 'anyPolicy' OID and having a critical policy constraints extension with the explicitPolicy field with value 0. The evaluator shall present the certificate to the TSF for authentication and observe that the TSF indicates the certificate is invalid.*
- *Test 20.6: The evaluator shall establish a certificate for the subject asserting the required policy but issued by a Certification Authority that does not include any certificate policy related extensions. The evaluator shall present the certificate for authentication and observe that the TSF indicates the certificate is invalid.*
- *Test 20.7: The evaluator shall establish a certificate for the subject asserting the required certificate policy issued by a Certification Authority that only asserts a single, non-matching policy OID in its policy related extensions (i.e., the CA certificate does not include the matching OID, 'anyPolicy' assertions or assert an OID that is mapped to the required OID via policy matching extensions by previous Certification Authorities in the certificate chain, if supported). The evaluator shall present the certificate to the TSF for authentication and observe the TSF indicates the certificate is invalid.*
- *Test 20.8: [conditional] If the ST indicates the inhibitAnyPolicy extension is supported, the evaluator shall establish a certificate for the subject asserting the required policy issued by a CA asserting the 'anyPolicy' OID, which is in turn issued by a CA with an inhibitAny extension with value 0. The evaluator shall present the certificate to the TSF for authentication and observe the TSF indicates the certificate is invalid.*

  Note that installing a root CA trusted by the TOE with the inhibitAny extension may be required if the TSF limits the path length of certificate chains.

5.2.5 Security Management (FMT)

FMT_SMF.1/AuthSvr Specification of Management Functions (Authentication Server)

FMT_SMF.1.1/AuthSvr

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

- *Ability to configure claimant verification data*
- *Ability to manage trust store data*
- *Ability to configure administrator authentication credential*
- *Ability to configure trusted channel to relying party*
- *[**selection**:*
  - *Ability to configure IPsec functionality*
  - *Ability to configure DTLS functionality*
  - *Ability to configure TLS functionality*
  - *Ability to manage claimant authentication policy*
  - *Ability to manage supported authentication-verification methods*
  - *Ability to manage supported authentication-verification methods supported via pass-through functionality*
  - *Ability to configure RADIUS shared secret*
  - *Ability to define authorized relying parties*
  - *Ability to configure cryptographic key storage*
  - *Ability to configure lockout policy for failed claimant authentication*
  - *Ability to unlock a claimant account*
  - *Ability to configure certificate validation checking mechanisms*
  - *Ability to define conditions in which claimant authentication attempts are rejected*
  - *Ability to associate pre-shared keys with claimants or external entities*
  - *Ability to configure restrictions on the composition of pre-shared keys*
  - *Ability to configure restrictions on the validation of pre-shared keys*
  - *Ability to generate pre-shared keys*
  - *Ability to accept pre-shared keys*
  - *Ability to manage HOTP verification function*
  - *Ability to manage TOTP verification function*
  - *No other functions*

  ]

].

Application Note: This SFR defines additional management functions for the TOE beyond what is defined in the Base-PP as FMT_SMF.1.

Evaluation Activities

FMT_SMF.1/AuthSvr
TSS
The evaluator shall verify that the TSS identifies all of the security-relevant management functions that apply to the security functions the TOE claims from this PP-Module.

Guidance
For each claimed management function, the evaluator shall ensure that the operational guidance contains instructions for how to configure the function.

Tests
- *Test 21: For each claimed management function, the evaluator shall follow the operational guidance to configure the behavior of that function and ensure that applying the configuration settings have the intended effect. Note that some or all of these functions may be tested in the course of performing the test activities for other claimed SFRs.*

5.2.6 TOE Access (FTA)

FTA_TSE.1 TOE Session Establishment

FTA_TSE.1.1

The TSF shall be able to deny **claimant** session establishment based on [*invalid certificate, [**selection**: [**assignment**: other identity attributes], no other attributes]*].

Application Note: The intent of this SFR is to describe any circumstances that would cause a claimant's authentication request to be rejected. All compliant TOEs will reject authentication requests based on invalid credentials. A compliant TOE may also impose additional limitations such as suspended accounts or time of day restrictions, depending on the capabilities of the TSF's authentication mechanism.

Evaluation Activities

FTA_TSE.1
TSS
The evaluator shall examine the TSS to determine that all of the attributes on which a claimant session can be denied are specifically defined.

Guidance
The evaluator shall examine the operational guidance to verify that it contains instructions for configuring each of the attributes identified in the TSS.

Tests
- *Test 22: The evaluator shall successfully have a claimant be authenticated by the TOE. For each attribute claimed in the SFR, the evaluator shall configure the TOE to deny user access based on a specific value of that attribute. The evaluator shall then attempt to establish a new session in contravention to the attribute setting while still providing valid authentication data. The evaluator shall observe that the access attempt fails.*

5.2.7 Trusted Path/Channels (FTP)

FTP_ITC.1/NAS Inter-TSF Trusted Channel (Relying Party Communications)

FTP_ITC.1.1/NAS

The TSF shall provide **[selection: *an IPsec, a RadSec, a mutually authenticated TLS, a mutually authenticated DTLS*]** communication channel between itself and **a relying party** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from modification or disclosure.

Application Note: If "an IPsec" is selected, the Base-PP SFR FCS_IPSEC_EXT.1 must be claimed.
If "a RadSec" is selected, the selection-based SFR FCS_RADSEC_EXT.1 must be claimed.

If "a mutually authenticated TLS" is selected, the Base-PP SFRs FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2 must be claimed.

If "a mutually authenticated DTLS" is selected, the Base-PP SFRs

FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2 must be claimed.

FTP_ITC.1.2/NAS

The TSF shall permit [*the TSF, **or the relying party***] to initiate communication via the trusted channel.

FTP_ITC.1.3/NAS

The TSF shall initiate communication via the trusted channel for [*responses to authentication request messages received from the relying party*].

Evaluation Activities

FTP_ITC.1/NAS
TSS
The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each communications mechanism is identified in terms of the allowed protocols for that IT entity. The evaluator shall also confirm that all protocols listed in the TSS are specified and included in the requirements in the ST.

Guidance
The evaluator shall confirm that the guidance documentation contains instructions for establishing and reestablishing the allowed protocols with each authorized IT entity.

Tests
- *Test 23: For each claimed trusted channel mechanism, the evaluator shall configure the TOE to interact with a relying party using that channel and verify using packet captures that the claimed mechanism is used.*

5.3 TOE Security Functional Requirements Rationale

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

Table 3: SFR Rationale

Objective	Addressed by	Rationale
O.AUTHORIZED_USE	FAU_GEN.1/AuthSvr	The TOE implements an audit mechanism to detect potential misuse of the TOE.
	FCS_CKM.3	The TOE implements a cryptographic key access control mechanism to prevent compromise of data in transit confidentiality mechanisms.
	FCS_STG_EXT.1	The TOE implements a cryptographic key storage mechanism to prevent compromise of data in transit confidentiality mechanisms.
	FIA_UAU.6	The TOE implements a re-authentication mechanism to prevent compromise of an administrator account to an unattended session.
	FMT_SMF.1/AuthSvr	The TOE implements management functions as a legitimate mechanism to control the behavior of the TSF.
O.SECURITY_ASSOCIATION	FCO_NRO.1	The TOE provides a non-repudiation function to assert the source of origin of its communications with the relying party.
	FCO_NRR.1	The TOE provides a proof of receipt function to assert to a relying part that communications with it are successful.
	FCS_EAPTLS_EXT.1	The TOE implements either EAP-TLS or EAP-TTLS as a mechanism to assert the success or failure of claimant authentication requests to the relying party.
O.TRUSTED_RP	FCS_EAPTLS_EXT.1	The TOE implements either EAP-TLS or EAP-TTLS with mutual authentication to authenticate itself to a relying party.
	FTP_ITC.1/NAS	The TOE implements a cryptographically secure trusted channel with a relying party that allows it to authorize itself to the relying party.

	FCS_RADSEC_EXT.1	RadSec is one potential mechanism by which the TOE can establish a trusted channel with the relying party.
	FIA_PSK_EXT.1/AuthSvr	Depending on the configuration of the trusted channel used to communicate with the relying party, the TOE may use a pre-shared key as the mechanism by which it authenticates itself to the relying party.
O.USER_AUTH	FCS_RADIUS_EXT.1	The TOE implements RADIUS, DIAMETER, or some other direct identity federation protocol to communicate the success or failure of claimant authentication requests.
	FIA_AFL.1/AuthSvr	The TOE implements a mechanism to prevent claimant authentication attempts if an excessive number of failed attempts have been made.
	FIA_X509_EXT.1/AuthSvr	The TOE implements a mechanism to determine the validity of X.509 certificates that are used as claimant authenticators.
	FTA_TSE.1	The TOE implements a mechanism to assert the failure of a claimant authentication attempt based on attributes of the claimant or of the attempt.
	FIA_HOTP_EXT.1	The TOE optionally supports HOTP as a form of claimant authenticator and implements mechanisms to determine the validity of a HOTP credential if supported.
	FIA_PSK_EXT.1/AuthSvr	The TOE optionally supports PSKs as a form of claimant authenticator and may specifically support one or more of HOTP, TOTP, or password-based PSK.
	FIA_PSK_EXT.2	The TOE optionally supports randomly generated PSKs as a form of claimant authenticator.
	FIA_PSK_EXT.3	The TOE optionally supports password-based PSKs as a form of claimant authenticator and implements mechanisms to determine the validity of a password-based PSK credential if supported.
	FIA_TOTP_EXT.1	The TOE optionally supports TOTP as a form of claimant authenticator and implements mechanisms to determine the validity of a TOTP credential if supported.

5.4 TOE Security Assurance Requirements

This PP-Module does not define any Security Assurance requirements. The SARs from the Base-PP must be satisfied.

6 Consistency Rationale

6.1 Collaborative Protection Profile for Network Devices

6.1.1 Consistency of TOE Type

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

6.1.2 Consistency of Security Problem Definition

PP-Module Threat, Assumption, OSP	Consistency Rationale
T.FALSE_ENDPOINTS	This threat is similar to the T.WEAK_AUTHENTICATION_ENDPOINTS threat in the NDcPP but it applies specifically to the NAS, which is an environmental component that is defined specifically in this PP-Module.
T.INVALID_USERS	This threat is similar to the T.UNAUTHORIZED_ADMINISTRATOR_ACCESS threat in the NDcPP but it applies to user authentication brokered by the TSF rather than to administrator authentication to the TOE itself. It is also similar to the T.UNTRUSTED_COMMUNICATION_CHANNELS threat in the NDcPP except that it applies specifically to the RADIUS communications and the protocols used to secure those, which is an interface that is defined specifically in this PP-Module.
A.RP_FEDERATION	The NDcPP does not define any assumptions for the intended network architecture that the TOE is deployed into. Therefore, an assumption that the network can be set up in such a way that the TOE will have direct connectivity with one or more relying parties does not violate any assumptions of the NDcPP.
P.AUTH_POLICY	This OSP relates to behavior that is not part of the Base-PP and so the Base-PP is not contradicted by guidance on its implementation.

6.1.3 Consistency of Objectives

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

PP-Module TOE Objective	Consistency Rationale
O.AUTHORIZED_USE	The NDcPP does not define any TOE objectives; instead, it maps SFRs directly to threats. This TOE objective is intended to mitigate the T.UNAUTHORIZED_ADMINISTRATOR_ACCESS and T.UNDETECTED_ACTIVITY threats for the functions defined by this PP-Module.
O.SECURITY_ASSOCIATION	The NDcPP does not define any TOE objectives; instead, it maps SFRs directly to threats. This objective is therefore assumed not to conflict with the NDcPP by virtue of the fact that the SFRs used to satisfy this objective do not conflict with the NDcPP SFRs.
O.TRUSTED_RP	The NDcPP does not define any TOE objectives; instead, it maps SFRs directly to threats. This objective is therefore assumed not to conflict with the NDcPP by virtue of the fact that the SFRs used to satisfy this objective do not conflict with the NDcPP SFRs.
O.USER_AUTH	The NDcPP does not define any TOE objectives; instead, it maps SFRs directly to threats. This objective is therefore assumed not to conflict with the NDcPP by virtue of the fact that the SFRs used to satisfy this objective do not conflict with the NDcPP SFRs.

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

PP-Module OE Objective	Consistency Rationale
OE.RP_FEDERATION	The Base-PP does not define where in a particular network architecture a network device must be deployed since it is designed to be generic to various types of network devices. This PP-Module defines the expected architectural deployment specifically for a network device that acts as an authentication server.
OE.REQUIRE_AUTH	The Base-PP does not define a particular use for a network device, so there is no consistency issue with the PP-Module defining expectations for the use of a specific type of device.

6.1.4 Consistency of Requirements

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

PP-Module Requirement	Consistency Rationale
Modified SFRs	
FIA_X509_EXT.1/Rev	This PP-Module modifies the Base-PP's definition of the SFR by making it mandatory rather than selection-based.
FIA_X509_EXT.2	This PP-Module modifies the Base-PP's definition of the SFR by making it mandatory rather than selection-based.
FIA_X509_EXT.3	This PP-Module modifies the Base-PP's definition of the SFR by making it mandatory rather than selection-based.
Additional SFRs	
This PP-Module does not add any requirements when the NDcPP is the base.	
Mandatory SFRs	
FAU_GEN.1/AuthSvr	This SFR iterates the FAU_GEN.1 SFR defined in the Base-PP to define auditable events for the functionality that is specific to this PP-Module.
FCO_NRO.1	This SFR applies to the implementation of the supported authentication protocol, which is beyond the original scope of the Base-PP.
FCO_NRR.1	This SFR applies to the implementation of the supported authentication protocol, which is beyond the original scope of the Base-PP.
FCS_CKM.3	The Base-PP requires confidentiality of cryptographic key data in FPT_SKP_EXT.1. This SFR defines more specific detail on how that function should be enforced.
FCS_EAPTLS_EXT.1	This SFR applies to the implementation of EAP-TLS; the Base-PP defines implementation requirements for (D)TLS, but EAP-TLS is beyond the original scope of the Base-PP.
FCS_RADIUS_EXT.1	This SFR applies to the implementation of authentication protocols, which is beyond the original scope of the Base-PP.
FCS_STG_EXT.1	This SFR is consistent with the FPT_SKP_EXT.1 requirement of the Base-PP but requires the TSF to implement a specific method of protecting key data rather than a general statement that such data is not stored in plaintext.
FIA_AFL.1/AuthSvr	This SFR defines functional behavior enforced on external users being authenticated by the TOE, which is functionality that is not covered by the Base-PP.
FIA_UAU.6	This SFR defines support for re-authentication of administrators, which expands on the authentication functionality defined in the Base-PP.
FIA_X509_EXT.1/AuthSvr	The Base-PP defines X.509 validation requirements for external entities presenting certificates to the TOE. This PP-Module defines a separate iteration of this function to define the certificate validation behavior that is enforced against claimants requesting to be authenticated by the TOE. It is substantially refined from its original definition to address issues specific to the handling of claimant certificates.
FMT_SMF.1/AuthSvr	This SFR defines additional management functionality that is specific to the PP-Module's product type and would therefore not be expected to be present in the Base-PP.
FTA_TSE.1	This SFR relates to the handling of claimants being authenticated by the TOE, which is functionality that is beyond the original scope of the Base-PP.
FTP_ITC.1/NAS	This SFR iterates the FTP_ITC.1 SFR defined in the Base-PP to define trusted communication channels for the functionality that is specific to this PP-Module.
Optional SFRs	

This PP-Module does not define any Optional requirements.	
Objective SFRs	
This PP-Module does not define any Objective requirements.	
Implementation-based SFRs	
This PP-Module does not define any Implementation-based requirements.	
Selection-based SFRs	
FCS_RADSEC_EXT.1	This SFR defines the implementation of RadSec and the peer authentication method that it uses. This relies on the TLS requirements defined by the Base-PP and may also use the X.509v3 certificate validation methods specified in the Base-PP, depending on the selected peer authentication method.
FIA_HOTP_EXT.1	This SFR extends the functionality of the Base-PP by defining the use of pre-shared keys as an authenticator.
FIA_PSK_EXT.1/AuthSvr	This SFR extends the functionality of the Base-PP by defining the use of pre-shared keys as an authenticator.
FIA_PSK_EXT.2	This SFR extends the functionality of the Base-PP by defining the use of pre-shared keys as an authenticator.
FIA_PSK_EXT.3	This SFR extends the functionality of the Base-PP by defining the use of pre-shared keys as an authenticator.
FIA_TOTP_EXT.1	This SFR extends the functionality of the Base-PP by defining the use of pre-shared keys as an authenticator.

Appendix A - Optional SFRs

A.1 Strictly Optional Requirements

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

A.2 Objective Requirements

This PP-Module does not define any Objective SFRs.

A.3 Implementation-based Requirements

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

Appendix B - Selection-based Requirements

B.1 Cryptographic Support (FCS)

FCS_RADSEC_EXT.1 RadSec

> ***The inclusion of this selection-based component depends upon selection in FTP_ITC.1.1/NAS.***

FCS_RADSEC_EXT.1.1

The TSF shall implement RadSec as specified in [**selection**: *RFC 6614*, *RFC 7360*] as updated by RFC 8996 to communicate securely with a relying party.

FCS_RADSEC_EXT.1.2

The TSF shall perform relying party authentication using X.509v3 certificates in accordance with FIA_X509_EXT.1**/Rev** and [**selection**: *pre-shared keys*, *no other methods*].

Application Note: It is recommended that both X.509v3 certificates and pre-shared keys be supported for resiliency purposes.

FCS_RADSEC_EXT.1.3

The TSF shall implement RadSec using [**selection**: *TLS in accordance with FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2* ***from the Base-PP***, *DTLS in accordance with FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2* ***from the Base-PP***] with mutual authentication.

Application Note: This SFR is claimed if "a RadSec" is selected in FTP_ITC.1.1/NAS.

If "RFC 6614" is claimed in the selection for FCS_RADSEC_EXT.1.1, "TLS in accordance with FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2 from the Base-PP" is claimed in FCS_RADSEC_EXT.1.3.

If "RFC 7360" is claimed in the selection for FCS_RADSEC_EXT.1.1, "DTLS in accordance with FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2 from the Base-PP" is claimed in FCS_RADSEC_EXT.1.3. Note that RFC 7360 is not directly updated by RFC 8996, but its references to DTLS 1.2 (RFC 6347) are.

It is the intent that TLS 1.0, TLS 1.1, and DTLS 1.0 (to include via downgrade as allowed in the original RFCs) are not allowed here.

If "pre-shared keys" is selected in FCS_RADSEC_EXT.1.2, the selection-based SFR FIA_PSK_EXT.1/AuthSvr must be claimed.

Evaluation Activities

FCS_RADSEC_EXT.1
TSS
The evaluator shall review the ST and verify that the RadSec protocol is described and that the description of the RadSec protocol includes a description of the support for relying party authentication in accordance with the requirement.

Guidance
If the ST indicates multiple options for relying party authentication, the evaluator shall review the operational guidance to ensure it includes instructions for configuring the options.

Tests
There are no test EAs for this component.

B.2 Identification and Authentication (FIA)

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

> ***The inclusion of this selection-based component depends upon selection in FCS_EAPTLS_EXT.1.3, FIA_PSK_EXT.1.2/AuthSvr.***

FIA_HOTP_EXT.1.1

The TSF shall support HMAC-Based One-Time Password authentication (HOTP) in accordance with RFC 4226.

FIA_HOTP_EXT.1.2

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

FIA_HOTP_EXT.1.3

The TSF shall generate a new HOTP seed key for each claimant to be authenticated.

FIA_HOTP_EXT.1.4

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

FIA_HOTP_EXT.1.5

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

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

].

FIA_HOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**:*
  - *administrator configurable value*
  - *[**assignment**: value less than 10]*

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

].

FIA_HOTP_EXT.1.7

The TSF shall not verify HOTP attempts outside of the counter look-ahead window of [**assignment**: *a value less than or equal to three*] [**selection**:

- *except for resynchronization*
- *where a look-ahead window of [**selection**: a configurable value, [**assignment**: fixed value]] is used to reset the counter but which is not considered a valid HOTP value*
- *with no exception*

].

FIA_HOTP_EXT.1.8

The TSF shall increment the counter after each successful authentication.

Application Note: This SFR is claimed if "HOTP" is selected in FCS_EAPTLS_EXT.1.3 or if "HMAC-based one-time password" is selected in FIA_PSK_EXT.1.2/AuthSvr.

The selection in FIA_HOTP_EXT.1.4 must be consistent with the key size specified for the size of the keys used in conjunction with the keyed-hash message authentication.

In FIA_HOTP_EXT.1.5, the ST author may either provide a configurable character length of at least 6 or a preset size between 6 and 10.

In FIA_HOTP_EXT.1.6, the ST may select throttle requests, account lockout, or both.

The selections in FIA_HOTP_EXT.1.7 indicate an optional resynchronization process that allows an arbitrary look-ahead to determine a new counter value by the verifier. This can be used for out-of-sync claimants or for initial use of a HOTP mechanism. The HOTP value presented for resynchronization does not serve to authenticate a user. A second verification using a small look-ahead window (at most three) is required after resynchronization to ensure the claimant and the TOE counter values are indeed synchronized so that arbitrary values are not considered valid. If such a resynchronization function is not supported, ‘with no exception’ is claimed.

The HOTP seed and all derived values are considered secret keys for purposes of protection.

Evaluation Activities

FIA_HOTP_EXT.1
TSS
The evaluator shall confirm the TSS describes how the TOE complies with the RFC.

The evaluator shall confirm the TSS describes how the HOTP seed is generated and ensure it aligns with FCS_RBG_EXT.1.

The evaluator shall confirm the TSS describes how the HOTP seed is protected and ensure it aligns with the key storage requirements in FCS_STG_EXT.1.

The evaluator shall confirm that the TSS describes HOTP seed keys and HOTP security parameters and indicates controlled access in accordance with FCS_CKM.3, and that it includes a description of encrypted export for seed keys that require transfer to claimant devices.

The evaluator shall confirm the TSS describes how a new HOTP seed is assigned for each claimant and how each claimant is uniquely identified.

The evaluator shall confirm the TSS describes how the HOTP seed is conditioned into a HOTP hash and verify it matches the selection in FIA_HOTP_EXT.1.4.

The evaluator shall confirm the TSS describes how the HOTP hash is truncated and verify it matches the selection in FIA_HOTP_EXT.1.5.

The evaluator shall confirm the TSS describes how the TOE handles multiple incoming invalid requests and verify it provides an anti-hammer mechanism that matches the selections made in FIA_HOTP_EXT.1.6.

The evaluator shall confirm the TSS describes how the TOE handles resynchronization and how it rejects values outside of the look-ahead window selected in FIA_TOTP_EXT.1.7 for claimant authentication.

Guidance
The evaluator shall verify the operational guidance contains all configuration guidance for setting any administrative value that is configurable in the FIA_HOTP_EXT.1 requirements.

Tests
The following tests may be performed in conjunction with those in FCS_EAPTLS_EXT.1. The evaluator shall follow any instructions in the operational guidance to establish a HOTP seed key for a registered claimant, initialize a claimant HOTP token, synchronize the counter between the token and the TSF if necessary, and perform the tests listed below.

Note that response of the TSF for an invalid HOTP value may be a request for the claimant to submit a new value, an indication that resynchronization is required, or an indication that the claimant's authentication request is rejected. Any of these responses is acceptable; a response that the authentication request is successful in Test 25 or Test 26 represents a failed test.

- *Test 24: The evaluator shall attempt to authenticate the claimant using a valid HOTP token and confirm that the TSF determines the value to be valid.*
- *Test 25: The evaluator shall advance a registered claimant's HOTP token counter beyond the look-ahead window and attempt to validate the claimant using the next HOTP token value. The evaluator shall observe that the TSF does not indicate the claimant is authenticated.*
- *Test 26: The evaluator shall replay a previous HOTP value from a registered claimant (e.g., the value used in Test 24) and confirm that the TSF does not indicate the claimant is authenticated.*
- *Test 27: [conditional on support for resynchronization] The evaluator shall repeat Test 25 and then follow the operational guidance to resynchronize the counter. The evaluator shall observe that after resynchronization, the TSF does not indicate successful authentication, but instead requests an additional HOTP value from the claimant.*

FIA_PSK_EXT.1 Pre-Shared Key Usage

FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for [**assignment**: *protocols or authentication schemes that use pre-shared keys*].

FIA_PSK_EXT.1.2

The TSF shall be able to accept the following as pre-shared keys: [**assignment**: *types of supported pre-shared keys (e.g., randomly generated, password-based, OTP)*].

Evaluation Activities

FIA_PSK_EXT.1

FIA_PSK_EXT.1/AuthSvr Pre-Shared Key Usage (Claimant Authentication)

The inclusion of this selection-based component depends upon selection in FCS_EAPTLS_EXT.1.3, FCS_RADSEC_EXT.1.2.

FIA_PSK_EXT.1.1/AuthSvr

The TSF shall be able to use pre-shared keys for [**selection**: *IPsec*, *RadSec*, *EAP-TTLS*, *RADIUS*, *HOTP*, *TOTP*].

FIA_PSK_EXT.1.2/AuthSvr

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

Application Note: This SFR is claimed if any other TOE functions require the use of pre-shared keys. Within the scope of this PP-Module, this includes the following:

- Any of "static PSK," "HOTP," or "TOTP" is selected in FCS_EAPTLS_EXT.1.3.
- "pre-shared keys" is selected in FCS_RADSEC_EXT.1.2.
- "pre-shared keys" is selected in FCS_IPSEC_EXT.1.13 (from the Base-PP).

IPsec is claimed in FIA_PSK_EXT.1.1/AuthSvr, if IPsec is claimed in FTP_ITC.1/NAS and the selection for FCS_IPSEC_EXT.1.13 in the Base-PP includes ‘pre-shared keys.’ PSK in IPsec may use any supported type of PSK – those supported are claimed in FIA_PSK_EXT.1.2/AuthSvr. Use of certificates in IPsec is preferred.

RadSec is claimed in FIA_PSK_EXT.1.1/AuthSvr, if RadSec is selected in FTP_ITC.1/NAS and the (D)TLS implementation in RadSec allows server-only authentication or supports a PSK ciphersuite. RadSec can use any type of PSK – those supported should be claimed in FIA_PSK_EXT.1.2/AuthSvr. Use of a mutually authenticated (D)TLS channel using certificate-based authentication is preferred.

If a pre-shared key is used in RADIUS to authenticate the relying party, RADIUS is claimed. It should not be claimed when RADIUS is used exclusively with RadSec since in that case, the legacy PSK used in RADIUS is replaced with a fixed value not used in authenticating the relying party. Use of a mutually authenticated channel using certificates to authenticate the relying party is preferred.

EAP-TTLS is claimed in FIA_PSK_EXT.1.1/AuthSvr if it is claimed in FCS_EAPTLS_EXT.1.1 and support for static PSK (alone or in combination) is indicated in FCS_EAPTLS_EXT.1.3. When EAP-TTLS is claimed in FIA_PSK_EXT.1.1/AuthSvr, at least one of ‘password-based,’ ‘HMAC-based one-time password,’ or ‘time-based one-time password’ is claimed in FIA_PSK_EXT.1.2/AuthSvr. Multiple password types are claimed if the TSF supports validation of combinations of password types, even if presented in a single payload.

Note that even if presented by a claimant, PSK are ignored in EAP-TLS implementations.

HOTP or TOTP, respectively, are claimed if the respective entry is claimed in FCS_EAPTLS_EXT.1.3 and the TSF validates the HOTP or TOTP values presented in an authentication request. If claimed, the PSK represents the seed key value generated by the TOE and shared via out-of-band mechanisms with the claimant as well as the HOTP or TOTP values presented by a claimant to be validated; the ‘generated as bit-based PSK’ as well as the respective HOTP or TOTP entries are claimed in FIA_PSK_EXT.1.2/AuthSvr. The selection-based SFRs FIA_HOTP_EXT.1 and FIA_TOTP_EXT.1 must also be claimed if HOTP or TOTP are claimed, respectively.

Note that if HOTP or TOTP mechanisms are supported, but the values are only validated by an external entity, the HOTP or TOTP entries are not claimed in FIA_PSK_EXT.1/AuthSvr.

If "generated bit-based" is selected in FIA_PSK_EXT.1.2/AuthSvr, FIA_PSK_EXT.2 must be claimed.

If "password-based" is selected in FIA_PSK_EXT.1.2/AuthSvr, FIA_PSK_EXT.3 must be claimed.

Evaluation Activities

FIA_PSK_EXT.1/AuthSvr
TSS
The evaluator shall examine the TSS to ensure that it identifies all protocols that allow pre-shared keys. For each protocol identified by the requirement, the evaluator shall confirm that the TSS states which pre-shared key selections are supported.

Guidance
The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on how to configure all selected pre-shared key options if any configuration is required.

Tests
The evaluator shall also perform the following tests for each protocol (or instantiation of a protocol, if performed by a different implementation on the TOE).

- *Test 28: For each mechanism selected in FIA_PSK_EXT.1.2/AuthSvr, the evaluator shall attempt to establish a connection and confirm that the connection requires the selected factors in the PSK to establish the connection.*

FIA_PSK_EXT.2 Generated Pre-Shared Keys

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

FIA_PSK_EXT.2.1

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

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

].

Application Note: This SFR is claimed if "generated bit-based" is selected in FIA_PSK_EXT.1.2/AuthSvr.

Generated PSKs are expected to be shared between components via an out-of-band mechanism.

Evaluation Activities

FIA_PSK_EXT.2
TSS
If "generate" is selected, the evaluator shall confirm that this process uses the RBG specified in FCS_RBG_EXT.1 and the output matches the size selected in FIA_PSK_EXT.2.1.

Guidance
The evaluator shall confirm the operational guidance contains instructions for generating or entering pre-shared keys (based on the selection made in FIA_PSK_EXT.2.1) for each protocol identified in the FIA_PSK_EXT.1.1/AuthSvr.

Tests
- *Test 29: For each selection made in FIA_PSK_EXT.2.1, the evaluator shall input and register the PSK (whether this involves generating the key or inputting it). The evaluator shall attempt to use the PSK (e.g., by making a claimant authentication attempt using the registered key) and confirm that the TSF indicates the PSK matches.*

FIA_PSK_EXT.3 Password-Based Pre-Shared Keys

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

***FIA_PSK_EXT.1.2/AuthSvr*.**

FIA_PSK_EXT.3.1

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

FIA_PSK_EXT.3.2

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

FIA_PSK_EXT.3.3

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

FIA_PSK_EXT.3.4

The TSF shall not accept PSKs failing to meet [**selection**: *an administrator-defined*, *a fixed*] password policy indicating the maximum PSK length consistent with FIA_PSK_EXT.3.1 and [**selection**: ***[assignment**: password policy indicating minimum length and types of characters required], no additional password constraints*].

FIA_PSK_EXT.3.5

The TSF shall generate all salts using an RBG that meets FCS_RBG_EXT.1 and with entropy corresponding to the key size selected for PBKDF in FIA_PSK_EXT.3.3.

FIA_PSK_EXT.3.6

The TSF shall require the PSK to be entered in accordance with the [**selection**: *user authentication policy*, *protocol authentication requirement*].

FIA_PSK_EXT.3.7

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

Application Note: This SFR is claimed if "password-based" is selected in FIA_PSK_EXT.1.2/AuthSvr.

For FIA_PSK_EXT.3.1, the ST author assigns the maximum size of the PSK it supports; it must support at least 64 characters.

For FIA_PSK_EXT.3.2, the ST author assigns any other supported characters; if there are no other supported characters, they should select "no other characters."

For FIA_PSK_EXT.3.3, the ST author selects the parameters based on the PBKDF used by the TSF.

For FIA_PSK_EXT.3.4, if the minimum length is settable, then the ST author chooses "a value settable by the administrator." If the minimum length is not settable, the ST author fills in the assignment with the minimum length the PSK must be. This requirement is to ensure bounds work properly.

For FIA_PSK_EXT.3.7, the ST author may select one, both, or neither of the functions in alignment with NIST SP 800-63B.

Evaluation Activities

FIA_PSK_EXT.3
TSS
The evaluator shall examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are used.

Support for length: The evaluator shall check to ensure that the TSS describes the allowable ranges for PSK lengths, and that at least 64 characters or a length defined by the platform may be specified by the user.

Support for character set: The evaluator shall check to ensure that the TSS describes the allowable character set and that it contains the characters listed in the SFR.

Support for PBKDF: The evaluator shall examine the TSS to ensure that the use of PBKDF2 is described and that the key sizes match that described by the ST author.

The evaluator shall check that the TSS describes the method by which the PSK is first encoded and then fed to the hash algorithm. The settings for the algorithm (padding, blocking, etc.) shall be described, and the evaluator shall verify that these are supported by the selections in this component as well as the selections concerning the hash function itself.

For the NIST SP 800-132-based conditioning of the PSK, the required evaluation activities will be performed when doing the evaluation activities for the appropriate requirements (FCS_COP.1/KeyedHash).

The evaluator shall confirm that the minimum length is described.

The ST author shall provide a description in the TSS regarding the salt generation. The evaluator shall confirm that the salt is generated using an RBG described in FCS_RBG_EXT.1.

[conditional] If password strength meter or password denylist is selected, the evaluator shall examine the TSS to ensure any password checking functionality provided by the TSF is described and contains details on how the function operates.

Guidance
The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on the composition of strong text-based pre-shared keys and (if the selection indicates keys of various lengths can be entered) that it provides information on the range of lengths supported. The guidance must specify the allowable characters for pre-shared keys in accordance with FIA_PSK_EXT.3.2.

Tests
If the TSF supports a configurable password policy, the evaluator shall follow the operational guidance to configure the password policy based on the selections made in FIA_PSK_EXT.3.4. If the password policy is not configurable, the evaluator will determine the password policy used by the TSF.

The evaluator shall perform the following tests:

- *Test 30: The evaluator shall compose a representative set of PSK meeting the TSF's password policy, to include one having the maximum length, one having the minimum length, and which, in aggregate, contain each of the supported special characters and each type of character supported. For each of the PSK, the evaluator shall demonstrate successful registration and use of the PSK.*
- *Test 31: For each of the password policy constraints indicated in FCS_PSK_EXT.3.4, the evaluator shall compose a password violating the constraint and demonstrate that the TSF rejects the PSK.*
- *Test 32: [conditional] If a password strength meter is supported, when performing Tests 1 and 2, the evaluator shall confirm that the TSF reflects the indicated strength of the PSK.*
- *Test 33: [conditional] If the TOE supports a password denylist, the evaluator shall enter a denylisted password and verify that the password is rejected or flagged as such.*

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

The inclusion of this selection-based component depends upon selection in FCS_EAPTLS_EXT.1.3, FIA_PSK_EXT.1.2/AuthSvr.

FIA_TOTP_EXT.1.1

The TSF shall support Time-Based One-Time Password (TOTP) authentication in accordance with RFC 6238.

FIA_TOTP_EXT.1.2

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

FIA_TOTP_EXT.1.3

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

FIA_TOTP_EXT.1.4

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

FIA_TOTP_EXT.1.5

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

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

].

FIA_TOTP_EXT.1.6

The TSF shall [**selection**:

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

].

FIA_TOTP_EXT.1.7

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

FIA_TOTP_EXT.1.8

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

FIA_TOTP_EXT.1.9

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

Application Note: This SFR is claimed if "TOTP" is selected in FCS_EAPTLS_EXT.1.3 or if "time-based one-time password" is selected in FIA_PSK_EXT.1.2/AuthSvr.

The selection FIA_TOTP_EXT.1.4 must be consistent with the key size specified for the size of the keys used in conjunction with the keyed-hash message authentication.

In FIA_TOTP_EXT.1.5, the ST author may either provide a configurable character length of at least 6 or a preset size between 6 and 10.

In FIA_TOTP_EXT.1.6, the ST author may select throttle requests, account lockout, or both.

The TOTP seed and all derived values are considered secret keys for purposes of protection.

Evaluation Activities

FIA_TOTP_EXT.1
TSS
The evaluator shall confirm the TSS describes how the TOE complies with the RFC.

The evaluator shall confirm the TSS describes how the TOTP seed is generated and ensure it aligns with FCS_RBG_EXT.1.

The evaluator shall confirm the TSS describes how the TOTP seed is protected and ensure it aligns with FCS_STG_EXT.1.

The evaluator shall confirm that the TSS describes TOTP seed keys and TOTP security parameters and indicates controlled access in accordance with FCS_CKM.3, and that it includes a description of encrypted export for seed keys that require transfer to claimant devices.

The evaluator shall confirm the TSS describes how a new TOTP seed is assigned for each claimant and how each claimant is uniquely identified.

The evaluator shall confirm the TSS describes how the TOTP seed is conditioned into a TOTP hash and verify that it matches the selection in FIA_TOTP_EXT.1.4.

The evaluator shall confirm the TSS describes how the TOTP hash is truncated and verify that it matches the selection in FIA_TOTP_EXT.1.5.

The evaluator shall confirm the TSS describes how the TOE handles multiple incoming invalid requests and verify it provides an anti-hammer mechanism that matches the selections in FIA_TOTP_EXT.1.6.

The evaluator shall confirm the TSS describes how the TOE handles resynchronization and how it rejects values outside of the look-ahead window selected in FIA_TOTP_EXT.1.7 for claimant authentication.

Guidance
The evaluator shall verify the operational guidance contains all configuration guidance for

setting any administrative value that is configurable in the FIA_TOTP_EXT.1 requirements.

Tests
The following tests may be performed in conjunction with those in FCS_EAPTLS_EXT.1. The evaluator shall follow any instructions in the operational guidance to establish a TOTP seed key for a registered claimant, initialize a claimant TOTP token, synchronize time between the token and the TSF if necessary, and perform the tests listed below.

Note that the response of the TSF for an invalid TOTP value may be a request for the claimant to submit a new value, or an indication that the claimant's authentication request is rejected. Any of these responses is acceptable; a response that the authentication request is successful in Test 35 or Test 36 represents a failed test.

- *Test 34: The evaluator shall attempt to authenticate the claimant using the TOTP token and confirm that the TSF determines the value to be valid.*
- *Test 35: The evaluator shall advance a registered claimant's TOTP token time beyond the allowed time drift and attempt to validate the claimant using the TOTP token value. The evaluator shall observe that the TSF does not indicate the claimant is authenticated.*
- *Test 36: The evaluator shall replay a previous TOTP value from a registered claimant (e.g., the value used in Test 34) after a delay equal to one time step and confirm that the TSF does not indicate the claimant is authenticated.*

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
Cryptographic Support (FCS)	FCS_EAPTLS_EXT EAP-TLS Protocol FCS_RADIUS_EXT Authentication Protocol FCS_RADSEC_EXT RadSec FCS_STG_EXT Cryptographic Key Storage
Identification and Authentication (FIA)	FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys FIA_PSK_EXT Pre-Shared Keys FIA_TOTP_EXT Time-Based One-Time Password Pre-Shared Keys

C.2 Extended Component Definitions

C.2.1 Cryptographic Support (FCS)

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

C.2.1.1 FCS_EAPTLS_EXT EAP-TLS Protocol

Family Behavior

This family defines requirements for how the TSF implements the Extensible Authentication Protocol (EAP) and EAP-Transport Layer Security.

Component Leveling

FCS_EAPTLS_EXT ——— 1

FCS_EAPTLS_EXT.1, EAP-TLS Protocol, requires the TSF to implement EAP and EAP-TLS according to appropriate standards.

Management: FCS_EAPTLS_EXT.1

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

- Configuration of claimant verification data
- Configuration of claimant authentication policy

Audit: FCS_EAPTLS_EXT.1

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

- Protocol failures
- Successful and failed authentication of claimant

FCS_EAPTLS_EXT.1 EAP-TLS Protocol

Hierarchical to: No other components.

Dependencies to: FCS_RBG_EXT.1 Random Bit Generation

[FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication, or

FCS_DTLSS_EXT.1 DTLS Server Protocol Without Mutual Authentication]

[FCS_TLSS_EXT.2 TLS Server Support for Mutual Authentication, or

FCS_DTLSS_EXT.2 DTLS Server Support for Mutual Authentication]

FIA_X509_EXT.1 X.509 Certificate Validation

FCS_EAPTLS_EXT.1.1

The TSF shall implement [**selection**: *EAP-TLS as specified in RFC 5216*, *EAP-TTLS as specified in RFC 5881*] as updated by RFC 8996 with [**selection**: *TLS*, *DTLS*] implemented using mutual authentication in

accordance with [**selection**: *FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2*, *FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2*].

FCS_EAPTLS_EXT.1.2

The TSF shall generate random values used in the [**selection**: *EAP-TLS*, *EAP-TTLS*] exchange using the RBG specified in FCS_RBG_EXT.1.

FCS_EAPTLS_EXT.1.3

The TSF shall support claimant authentication using certificates and [**selection**: *static PSK*, *HOTP*, *TOTP*, *other authentication-verification methods via pass-through*, *no other methods*].

FCS_EAPTLS_EXT.1.4

The TSF shall not forward an EAP-Success response to the relying party if the client certificate is not valid according to FIA_X509_EXT.1, if the [**selection**: *TLS*, *DTLS*] session is not established, or if any of [**selection**: *PSK*, *HOTP value*, *TOTP value*, *no other authenticator*] required by the authentication policy are not provided or if any of the required authenticators presented in the authentication request is not valid.

C.2.1.2 FCS_RADIUS_EXT Authentication Protocol

Family Behavior

Components in this family define requirements for implementation of authentication protocols.

Component Leveling

FCS_RADIUS_EXT ——— 1

FCS_RADIUS_EXT.1, Authentication Protocol, requires the TSF to implement the specified authentication protocols.

Management: FCS_RADIUS_EXT.1

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

- Ability to configure RADIUS shared secret
- Ability to define authorized NAS

Audit: FCS_RADIUS_EXT.1

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

- Protocol failures
- Success/failure of authentication

FCS_RADIUS_EXT.1 Authentication Protocol

Hierarchical to: No other components.

Dependencies to: FCS_EAPTLS_EXT.1 EAP-TLS Protocol

FCS_RADIUS_EXT.1.1

The TSF shall implement the [**selection**: *RADIUS protocol as specified in RFC 2865*, *DIAMETER protocol as specified in RFC 6733*, *[**assignment**: other direct identity federation protocol]*] for communication of identity and authentication information with a relying party.

FCS_RADIUS_EXT.1.2

The TSF shall implement encapsulated EAP in accordance with FCS_EAPTLS_EXT.1.

FCS_RADIUS_EXT.1.3

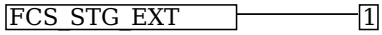
The TSF shall provide [**selection**: *a key indicator*, *an encrypted parameter*, *an encrypted value*] for a key held by the successfully authenticated claimant derived from the supported EAP mode and provided to the relying party in accordance with the protocol indicated in FCS_RADIUS_EXT.1.1.

C.2.1.3 FCS_STG_EXT Cryptographic Key Storage

Family Behavior

Components in this family define requirements for secure storage of cryptographic keys.

Component Leveling



FCS_STG_EXT.1, Cryptographic Key Storage, requires the TSF to identify a mechanism used to securely store cryptographic keys.

Management: FCS_STG_EXT.1

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

- Configuration of cryptographic key storage

Audit: FCS_STG_EXT.1

There are no auditable events foreseen.

FCS_STG_EXT.1 Cryptographic Key Storage

Hierarchical to: No other components.

Dependencies to: None

FCS_STG_EXT.1.1

Persistent private and secret keys shall be stored within the TSF [**selection**:

- *encrypted within a hardware protected key*
- *in a hardware cryptographic module*
- *within an isolated execution environment protected by a hardware key*

].

C.2.1.4 FCS_RADSEC_EXT RadSec

Family Behavior

Components in this family define requirements for the TSF to use RadSec to secure RADIUS data in transit.

Component Leveling

FCS_RADSEC_EXT.1, RadSec, defines implementation requirements for RadSec.

Management: FCS_RADSEC_EXT.1

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

- Configuration of trusted channel to relying party

Audit: FCS_RADSEC_EXT.1

There are no auditable events foreseen.

FCS_RADSEC_EXT.1 RadSec

Hierarchical to: No other components.

Dependencies to: FCS_RADIUS_EXT.1 Authentication Protocol

FCS_RBG_EXT.1 Random Bit Generation

[FIA_PSK_EXT.1 Pre-Shared Key Composition, or

FIA_X509_EXT.1 X.509 Certificate Validation]

FCS_RADSEC_EXT.1.1

The TSF shall implement RadSec as specified in [**selection**: *RFC 6614*, *RFC 7360*] as updated by RFC 8996 to communicate securely with a relying party.

FCS_RADSEC_EXT.1.2

The TSF shall perform relying party authentication using X.509v3 certificates in accordance with FIA_X509_EXT.1 and [**selection**: *pre-shared keys*, *no other methods*].

FCS_RADSEC_EXT.1.3

The TSF shall implement RadSec using [**selection**: *TLS in accordance with FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2*, *DTLS in accordance with FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2*] with mutual authentication.

C.2.2 Identification and Authentication (FIA)

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

C.2.2.1 FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys

Family Behavior

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

Component Leveling

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

Management: FIA_HOTP_EXT.1

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

- Ability to configure restrictions on the composition of pre-shared keys
- Ability to configure restrictions on the validation of pre-shared keys

Audit: FIA_HOTP_EXT.1

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

- Generation of a HOTP seed key
- Entity HOTP value comparison

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

Hierarchical to: No other components.

Dependencies to: FCS_COP.1 Cryptographic Operation

FCS_RBG_EXT.1 Random Bit Generation

FIA_HOTP_EXT.1.1

The TSF shall support HMAC-Based One-Time Password authentication (HOTP) in accordance with RFC 4226.

FIA_HOTP_EXT.1.2

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

FIA_HOTP_EXT.1.3

The TSF shall generate a new HOTP seed key for each claimant to be authenticated.

FIA_HOTP_EXT.1.4

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

FIA_HOTP_EXT.1.5

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

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

].

FIA_HOTP_EXT.1.6

The TSF shall [**selection**:

- *throttle invalid requests to [**selection**:*
  - *administrator configurable value*
  - *[**assignment**: value less than 10]*

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

].

FIA_HOTP_EXT.1.7

The TSF shall not verify HOTP attempts outside of the counter look-ahead window of [**assignment**: *a value less than or equal to three*] [**selection**:

- *except for resynchronization*
- *where a look-ahead window of [**selection**: a configurable value, [**assignment**: fixed value]] is used to reset the counter but which is not considered a valid HOTP value*
- *with no exception*

].

FIA_HOTP_EXT.1.8

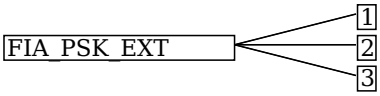
The TSF shall increment the counter after each successful authentication.

C.2.2.2 FIA_PSK_EXT Pre-Shared Keys

Family Behavior

Components in this family describe the requirements for pre-shared keys used for authentication.

Component Leveling



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

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

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

Management: FIA_PSK_EXT.1

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

- Ability to associate pre-shared keys with claimants or external entities

Audit: FIA_PSK_EXT.1

There are no auditable events foreseen.

FIA_PSK_EXT.1 Pre-Shared Key Usage

Hierarchical to: No other components.

Dependencies to: [FCS_EAPTLS_EXT.1 EAP-TLS Protocol, or

FCS_IPSEC_EXT.1 IPsec]

FIA_PSK_EXT.1.1

The TSF shall be able to use pre-shared keys for [**assignment**: *protocols or authentication schemes that use pre-shared keys*].

FIA_PSK_EXT.1.2

The TSF shall be able to accept the following as pre-shared keys: [**assignment**: *types of supported pre-shared keys (e.g., randomly generated, password-based, OTP)*].

Management: FIA_PSK_EXT.2

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

- Ability to generate pre-shared keys
- Ability to accept pre-shared keys

Audit: FIA_PSK_EXT.2

There are no auditable events foreseen.

FIA_PSK_EXT.2 Generated Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FIA_PSK_EXT.1 Pre-Shared Key Usage

FIA_PSK_EXT.2.1

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

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

].

Management: FIA_PSK_EXT.3

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

- Ability to configure restrictions on the composition of pre-shared keys
- Ability to configure restrictions on the validation of pre-shared keys

Audit: FIA_PSK_EXT.3

No auditable events are foreseen.

FIA_PSK_EXT.3 Password-Based Pre-Shared Keys

Hierarchical to: No other components.

Dependencies to: FCS_COP.1 Cryptographic Operation

FCS_RBG_EXT.1 Random Bit Generation

FIA_PSK_EXT.1 Pre-Shared Key Usage

FIA_PSK_EXT.3.1

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

FIA_PSK_EXT.3.2

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

FIA_PSK_EXT.3.3

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

FIA_PSK_EXT.3.4

The TSF shall not accept PSKs failing to meet [**selection**: *an administrator-defined*, *a fixed*] password policy indicating the maximum PSK length consistent with FIA_PSK_EXT.3.1 and [**selection**: *[**assignment**: password policy indicating minimum length and types of characters required], no additional password constraints*].

FIA_PSK_EXT.3.5

The TSF shall generate all salts using an RBG that meets FCS_RBG_EXT.1 and with entropy corresponding to the key size selected for PBKDF in FIA_PSK_EXT.3.3.

FIA_PSK_EXT.3.6

The TSF shall require the PSK to be entered in accordance with the [**selection**: *user authentication policy*, *protocol authentication requirement*].

FIA_PSK_EXT.3.7

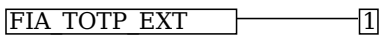
The TSF shall [**selection**: *provide a password strength meter*, *check the password against a denylist*, *perform no action to assist the user in choosing a strong password*].

C.2.2.3 FIA_TOTP_EXT Time-Based One-Time Password Pre-Shared Keys

Family Behavior

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

Component Leveling



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

Management: FIA_TOTP_EXT.1

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

- Ability to configure restrictions on the composition of pre-shared keys
- Ability to configure restrictions on the validation of pre-shared keys

Audit: FIA_TOTP_EXT.1

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

- Generation of a TOTP seed key
- Entity TOTP value comparison

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

Hierarchical to: No other components.

Dependencies to: FCS_COP.1 Cryptographic Operation

FCS_RBG_EXT.1 Random Bit Generation

FIA_TOTP_EXT.1.1

The TSF shall support Time-Based One-Time Password (TOTP) authentication in accordance with RFC 6238.

FIA_TOTP_EXT.1.2

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

FIA_TOTP_EXT.1.3

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

FIA_TOTP_EXT.1.4

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

FIA_TOTP_EXT.1.5

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

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

].

FIA_TOTP_EXT.1.6

The TSF shall [**selection**:

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

].

FIA_TOTP_EXT.1.7

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

FIA_TOTP_EXT.1.8

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

FIA_TOTP_EXT.1.9

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

Appendix D - Implicitly Satisfied Requirements

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

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

This PP-Module has no implicitly satisfied requirements. All SFR dependencies are explicitly met either through SFRs defined by the PP-Module or inherited from the Base-PP.

Appendix E - Allocation of Requirements in Distributed TOEs

For a distributed TOE, the SFRs in this PP-Module need to be met by the TOE as a whole, but not all SFRs will necessarily be implemented by all components. The following categories are defined in order to specify when each SFR must be implemented by a component:

- **All Components ("All"):** All components that comprise the distributed TOE must independently satisfy the requirement.
- **At least one Component ("One"):** This requirement must be fulfilled by at least one component within the distributed TOE.
- **Feature Dependent ("Feature Dependent"):** These requirements will only be fulfilled where the feature is implemented by the distributed TOE component (note that the requirement to meet the PP-Module as a whole requires that at least one component implements these requirements if they are claimed by the TOE).

The table below specifies how each of the SFRs in this PP-Module must be met, using the categories above.

Requirement	Description	Distributed TOE SFR Allocation
FAU_GEN.1/AuthSvr	Audit Data Generation (Authentication Server)	All
FCO_NRO.1	Selective Proof of Origin	Feature Dependent
FCO_NRR.1	Selective Proof of Receipt	Feature Dependent
FCS_CKM.3	Cryptographic Key Access	All
FCS_EAPTLS_EXT.1	EAP-TLS Protocol	Feature Dependent
FCS_RADIUS_EXT.1	Authentication Protocol	Feature Dependent
FCS_STG_EXT.1	Cryptographic Key Storage	All
FIA_AFL.1/AuthSvr	Authentication Failure Handling (Claimant)	Feature Dependent
FIA_X509_EXT.1/AuthSvr	X.509 Certificate Validation (Claimant)	Feature Dependent
FIA_UAU.6	Re-Authenticating	Feature Dependent
FMT_SMF.1/AuthSvr	Specification of Management Functions (Authentication Server)	All
FTA_TSE.1	TOE Session Establishment	Feature Dependent
FTP_ITC.1/NAS	Inter-TSF Trusted Channel (Relying Party Communications)	Feature Dependent
FCS_RADSEC_EXT.1 (selection-based)	RadSec	Feature Dependent
FIA_HOTP_EXT.1 (selection-based)	HMAC-Based One-Time Password Pre-Shared Keys	Feature Dependent
FIA_PSK_EXT.1/AuthSvr (selection-based)	Pre-Shared Key Usage	Feature Dependent
FIA_PSK_EXT.2 (selection-based)	Generated Pre-Shared Keys	Feature Dependent
FIA_PSK_EXT.3 (selection-based)	Password-Based Pre-Shared Keys	Feature Dependent
FIA_TOTP_EXT.1 (selection-based)	Time-Based One-Time Password Pre-Shared Keys	Feature Dependent

Appendix F - Entropy Documentation and Assessment

The TOE does not require any additional supplementary information to describe its entropy sources beyond the requirements outlined in the Base-PP.

Appendix G - Acronyms

Acronym	Meaning
AAA	Authentication, Authorization, and Accounting
Base-PP	Base Protection Profile
CC	Common Criteria
CEM	Common Evaluation Methodology
cPP	Collaborative Protection Profile
CRL	Certificate Revocation List
CSP	Critical Security Parameters
DTLS	Datagram Transport Layer Security
EAP	Extensible Authentication Protocol
HOTP	Hash-Based One-Time Password
IPsec	Internet Protocol Security
MSK	Master Session Key
OCSP	Online Certificate Status Protocol
OE	Operational Environment
PBKDF	Password-Based Key Derivation Function
PP	Protection Profile
PP-Configuration	Protection Profile Configuration
PP-Module	Protection Profile Module
PSK	Pre-Shared Key
RADIUS	Remote Authentication Dial In User Service
RBG	Random Bit Generator
RP	Relying Party
SAR	Security Assurance Requirement
SFR	Security Functional Requirement
SSH	Secure Shell
ST	Security Target
TLS	Transport Layer Security
TOE	Target of Evaluation
TOTP	Time-Based One-Time Password
TSF	TOE Security Functionality
TSFI	TSF Interface
TSS	TOE Summary Specification
WLAN	Wireless Local Area Network

Appendix H - 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.
[NDcPP]	collaborative Protection Profile for Network Devices, Version 2.2e, March 23, 2020
[NDcPP SD]	Supporting Document - Evaluation Activities for Network Device cPP, Version 2.2, December 2019