# PP-Module for Authentication Servers

Version: 2.0

2026-03-10

**National Information Assurance Partnership**

## Revision History

| Version | Date | Comment |
|---|---|---|
| 1.0 | 2023-01-25 | Initial Release |
| 2.0 | 2026-03-10 | Incorporate NIAP Technical Decisions, Update to CC:2022 |

## 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:

- Network Device collaborative Protection Profile Version 4.0

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 (CEM) | Common Evaluation Methodology for Information Technology Security Evaluation. |
| Direct Rationale | A type of Protection Profile, PP-Module, or Security Target in which the security problem definition (SPD) elements are mapped directly to the SFRs and possibly to the security objectives for the operational environment. There are no security objectives for the TOE. |
| 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).

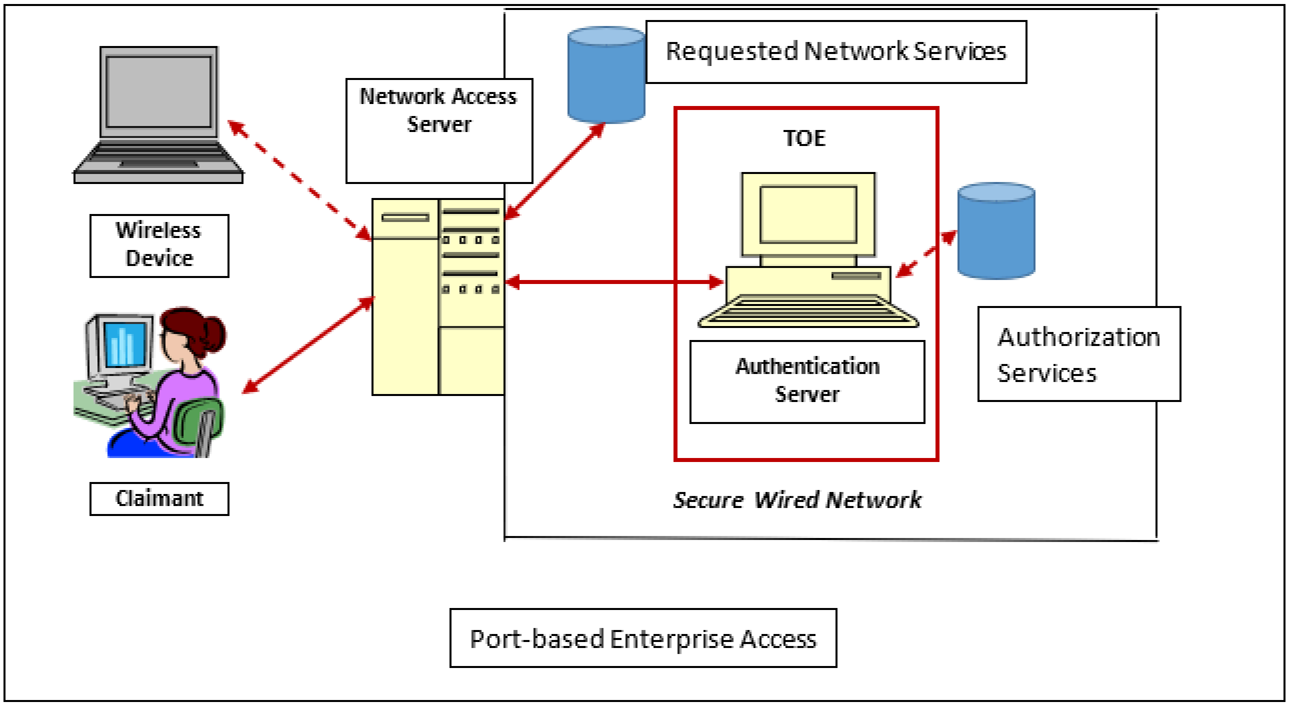


**Figure 1: NAS with an Authentication Server**

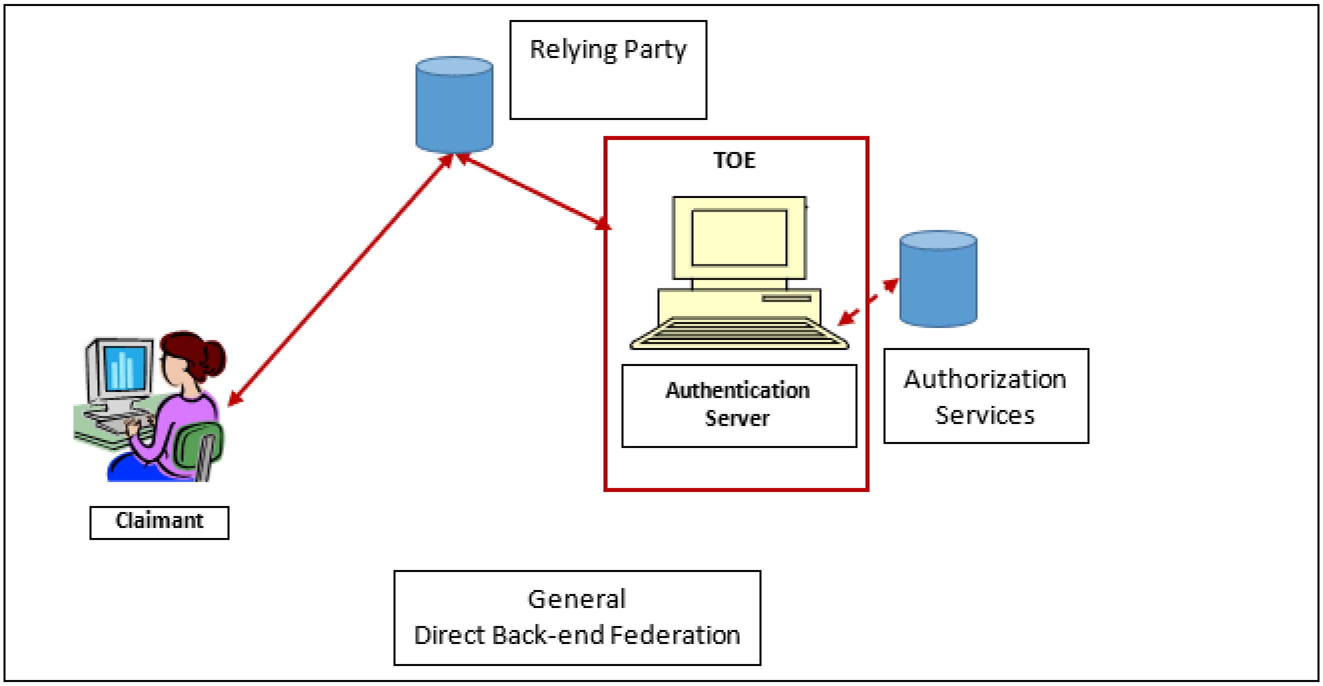


**Figure 2: Generic Authentication Server User Case**

## 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.

## 1.6 Package Usage

This section contains selections and assignments that are required when the listed Functional Packages are claimed by this PP-Module.
Package Usage guidance defined in the TOE's relevant Base-PP applies to the usage of the packages for this module, unless explicitly stated otherwise in this section.

### Functional Package for X.509, Version 1.0

**No CA Claims Permitted**

The TOE will not be a certificate authority because neither the Base-PP nor the PP-Module provides for the operation of an authentication server as a CA. Thus, the ST author shall select the option to request a certificate from an external CA in FIA_XCU_EXT.2.1 and shall not select any options elsewhere in the package that involve claiming the ability to be a CA.

**Certificate Verification and Assertion Required in FIA_XCU_EXT.1.1**

The ST author shall select the options to verify and assert certificate identities in FIA_XCU_EXT.1.1.

**Limitations on Signature Algorithms in FIA_X509_EXT.1.1**

The TOE must utilize appropriate cryptographic algorithms that conform to CNSA standards. Thus, the TOE shall utilize no other algorithms outside of those specified in RFC 8603 for certificate or CRL signatures. Additionally, the TOE shall not use ECDSA with SHA-512 signatures for OCSP responses, and shall utilize no other algorithms for OCSP responses.

**Required Extension Processing in FIA_X509_EXT.1.2**

The ST author shall select the option to process the basicConstraints and extendedKeyUsage extensions in FIA_X509_EXT.1.2. If the TOE supports the use of certificate names of types other than a Directory Name, the ST author shall select the option to process the subjectAlternateName extension and the relevant options within the subjectAlternateName extension that correspond to the supported identifier types. If the TOE supports the use of UPN names, the ST author shall utilize the assignment for other name types within the selection for extendedKeyUsage support to specify the use of the UPN name type.
The ST author shall utilize the assignment for other extensions within the list of selectable certificate extensions to specify supported policy-related extensions, including certificate policy, policy mapping, policy constraints, and inhibit anyPolicy.

In addition, the ST author shall perform the following tests as appropriate:

- *Test FIA_X509_EXT.1:AuthSvr:1: The evaluator shall test the following name constraints:*
  - *Test FIA_X509_EXT.1:AuthSvr:1.1: 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 FIA_X509_EXT.1:AuthSvr:1.2: For each leaf certificate used in Test FIA_X509_EXT.1:AuthSvr: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 FIA_X509_EXT.1:AuthSvr:1 for the new certificate and observe that the TSF indicates the certificate is invalid.*
  - *Test FIA_X509_EXT.1:AuthSvr:1.3: For each leaf certificate used in Test FIA_X509_EXT.1:AuthSvr: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 FIA_X509_EXT.1:AuthSvr:1 using the new certificate and observe that the TSF indicates the certificate is invalid.*
- *Test FIA_X509_EXT.1:AuthSvr:2: [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 FIA_X509_EXT.1:AuthSvr:2.1: 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 FIA_X509_EXT.1:AuthSvr:2.2: [conditional] If the ST selects 'Inhibit anyPolicy extension...," the evaluator shall repeat Test FIA_X509_EXT.1:AuthSvr:2.1 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 FIA_X509_EXT.1:AuthSvr:2.3: [conditional] If the ST indicates support for the policy mapping extension, the evaluator shall repeat Test FIA_X509_EXT.1:AuthSvr:2.1 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 FIA_X509_EXT.1:AuthSvr:2.4: [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 FIA_X509_EXT.1:AuthSvr:2.5: [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 FIA_X509_EXT.1:AuthSvr:2.6: 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 FIA_X509_EXT.1:AuthSvr:2.7: 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 FIA_X509_EXT.1:AuthSvr:2.8: [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.*

### CRL or OCSP-based Revocation Required for FIA_X509_EXT.1.3

The TOE must support revocation that only involves CRL or OCSP. Accordingly, the ST author shall select only from options involving CRL or OCSP in FIA_X509_EXT.1.3 (e.g., the selection to treat all certificates older than a given short timeframe is not an acceptable substitute or alternative for supporting CRL or OCSP).

### Connections to CRL or OCSP Servers Required for FIA_X509_EXT.1.4

Because the TOE is required to support CRL or OCSP, the TSF shall support an appropriate mechanism for obtaining revocation status information. In the case of CRL, the ST author shall claim that revocation status information is obtained via network connection to a CRL distribution point. In the case of OCSP, the ST author shall claim that revocation status information is obtained via network connection to an OCSP responder, via OCSP stapling, or via OCSP multi-stapling.

### Restrictions on Acceptable Key Usage Values for FIA_X509_EXT.1.5

The TOE will always support the use of extendedKeyUsage values to verify that X.509 certificates are used in accordance with their intended purpose. Accordingly, the ST author shall claim that the TOE supports the processing of extendedKeyUsage fields in the leaf certificate (as opposed to application of trust store context rules or passing the certification path or other supported context information to an external function) and shall select all values that are relevant to the claimed uses of X.509 in the ST. In particular, since neither the Base-PP nor the PP-Module does not define any functions that require the use of S/MIME, the ST author shall not select this as an extendedKeyUsage value to be validated.

### Required Function Claims in FIA_X509_EXT.2.1

The ST author shall ensure that the selections and assignments within FIA_X509_EXT.2.1 reflect the usage of X.509 for EAP-TLS. Other selections and assignments may be made as appropriate for other TOE functionality.

## Functional Package for Transport Layer Security (TLS), Version 2.1

### TLS or DTLS Client Functionality Required in FCS_TLS_EXT.1.1

The ST author shall select the option to support client functionality for TLS or DTLS in FCS_TLS_EXT.1.1. The selection shall correspond with other protocol selections made in FCS_EAPTLS_EXT.1.

### Mutual Authentication Required in FCS_TLSC_EXT.1.1 or FCS_DTLSC_EXT.1.1

The ST author shall select the option to support mutual authentication for TLS or DTLS, according with the protocol support selected in FCS_TLS_EXT.1.1.

# 2 Conformance Claims

### Conformance Statement

An ST must claim exact conformance to this PP-Module.

The evaluation methods used for evaluating the TOE are a combination of the workunits defined in [CEM] as well as the Evaluation Activities for ensuring that individual SFRs and SARs have a sufficient level of supporting evidence in the Security Target and guidance documentation and have been sufficiently tested by the laboratory as part of completing ATE_IND.1. Any functional packages this PP claims similarly contain their own Evaluation Activities that are used in this same manner.

### CC Conformance Claims

This PP-Module is conformant to Part 2 (extended) and Part 3 (conformant) of Common Criteria CC:2022, Revision 1.

### PP Claim

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

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

- Network Device collaborative Protection Profile Version 4.0

### Package Claim

- This PP-Module is Functional Package for TLS, version 2.1 conformant.
- This PP-Module is Functional Package for X.509, version 1.0 conformant.
- This PP-Module does not conform to any assurance packages.

The functional packages to which the PP conforms may include SFRs that are not mandatory to claim for the sake of conformance. An ST that claims one or more of these functional packages may include any non-mandatory SFRs that are appropriate to claim based on the capabilities of the TSF and on any triggers for their inclusion based inherently on the SFR selections made.

# 3 Security Problem Definition

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.

**T.UNAUTHORIZED_ADMINISTRATOR_ACCESS (from NDcPP)**

This threat from the above Base PP also applies to the functionality defined in this PP-Module.

**T.UNDETECTED_ACTIVITY (from NDcPP)**

This threat from the above Base PP also applies to the functionality defined in this PP-Module.

## 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

**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 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.

## 4.2 Security Objectives Rationale

This section describes how the assumptions and organizational security policies map to operational environment security objectives.

**Table 1: Security Objectives Rationale**

| Assumption or OSP | Security Objectives | Rationale |
|---|---|---|
| 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 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 Collaborative Protection Profile for Network Device 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 Network Device 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

This PP-Module does not modify any SFRs defined by the NDcPP.

## 5.2 TOE Security Functional Requirements

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

### 5.2.1 Auditable Events for Mandatory SFRs

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

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FAU_GEN.1/AuthSvr | No events specified | N/A |
| FCO_NRO.1 | Claimant request for which the TOE does not have credential verification data. | Identity of the claimant. |
| FCO_NRR.1 | No events specified | N/A |

| | | |
|---|---|---|
| FCS_CKM.3 | **[selection:** *Attempt to export plaintext key or CSP via defined interface., None***]** | If attempt is detected, record process identifier, authorized user's identifier (if any). |
| FCS_EAPTLS_EXT.1 | Successful and failed authentication of claimant. | Identifier of claimant. |
| | Protocol failures. | If failure occurs, record a descriptive reason for the failure. |
| FCS_RADIUS_EXT.1 | Protocol failures. | If failure occurs, record a descriptive reason for the failure. |
| | Success/failure of authentication. | No additional information |
| FCS_STG_EXT.1 | No events specified | N/A |
| 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. | No additional information |
| 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. |

## 5.2.2 Security Audit (FAU)

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

FAU_GEN.1.1/AuthSvr

The TSF shall be able to generate audit data 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 for Mandatory SFRs table (Table 2)*
d. *[**selection**: Auditable events listed in the Auditable Events for Selection-Based SFRs table (Table 7), no other events]*

].

**Application Note:** The auditable events defined in the audit tables 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.

If the ST includes any selection-based SFRs, the selection for "Auditable events listed in the Auditable Events for Selection-Based SFRs table" must be made. If no selection-based SFRs are included, "no other events" should be selected. The auditing of selection-based SFRs is only required if that SFR is included in the ST.

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

For FCS_CKM.3, if no defined interfaces have access to persistent keys or CSP, select 'none'.

FAU_GEN.1.2/AuthSvr

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

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

## 5.2.3 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*].

### 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*].

## 5.2.4 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.

### 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* ***as defined in the Functional Package for Transport Layer Security (TLS), version 2.1***, *FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2* ***as defined in the Functional Package for Transport Layer Security (TLS), version 2.1***].

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.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**as defined in Functional Package for X.509, version 1.0**, 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 Functional Package for Transport Layer Security (TLS), version 2.1 are claimed.

FCS_TLSS_EXT.1, FCS_TLSS_EXT.2, FCS_DTLSS_EXT.1, and FCS_DTLSS_EXT.2 are defined in the Functional Package for Transport Layer Security (TLS), version 2.1. 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.

### 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.

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

].

## 5.2.5 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.

### 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]*].

## 5.2.6 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.

## 5.2.7 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.

## 5.2.8 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 Functional Package for Transport Layer Security (TLS), version 2.1 SFRs FCS_TLSS_EXT.1 and FCS_TLSS_EXT.2 must be claimed.

If "a mutually authenticated DTLS" is selected, the Functional Package for Transport Layer Security (TLS), version 2.1 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*].

## 5.3 TOE Security Functional Requirements Rationale

The following rationale provides justification for each SFR for the TOE, showing that the SFRs are suitable to address the specified threats:

**Table 3: SFR Rationale**

| Threat | Addressed by | Rationale |
|---|---|---|
| T.FALSE_ENDPOINTS | FIA_X509_EXT.1 (from Functional Package for X.509, version 1.0) | Mitigates the threat by authenticating relying parties via X.509 certificates. |
| | FIA_X509_EXT.2 (from Functional Package for X.509, version 1.0) | Mitigates the threat by authenticating relying parties via X.509 certificates. |
| | FIA_X509_EXT.3 (from Functional Package for X.509, version 1.0) | Mitigates the threat by implementing a method for the TOE to obtain its own certificate for mutually-authenticated TLS or DTLS. |
| | FCS_EAPTLS_EXT.1 | Mitigates the threat by utilizing either EAP-TLS or EAP-TTLS with mutual authentication to authenticate itself to a relying party. |
| | FTP_ITC.1/NAS | Mitigates the threat by utilizing a cryptographically secure trusted channel with a relying party that allows mutual authentication with the relying party. |
| | FCS_RADSEC_EXT.1 (selection-based) | Mitigates the threat by utilizing RadSec to establish a trusted channel with the relying party. |
| | FIA_PSK_EXT.1/AuthSvr (selection-based) | Mitigates the threat by using a pre-shared key as the mechanism by which it authenticates itself to the relying party. |
| T.INVALID_USERS | FCO_NRO.1 | Mitigates the threat by providing a non-repudiation function to assert the source of origin of its communications with the relying party. |

| | | |
|---|---|---|
| | FCO_NRR.1 | Mitigates the threat by providing a proof of receipt function to assert to a relying part that communications with it are successful. |
| | FCS_EAPTLS_EXT.1 | Mitigates the threat by utilizing either EAP-TLS or EAP-TTLS as a mechanism to assert the success or failure of claimant authentication requests to the relying party. |
| | FCS_RADIUS_EXT.1 | Mitigates the threat by implementing RADIUS, DIAMETER, or some other direct identity federation protocol to communicate the success or failure of claimant authentication requests. |
| | FIA_AFL.1/AuthSvr | Mitigates the threat by implementing a mechanism to prevent claimant authentication attempts if an excessive number of failed attempts have been made. |
| | FTA_TSE.1 | Mitigates the threat by implementing 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 (selection-based) | Mitigates the threat by utilizing HOTP as a form of claimant authenticator and implementing mechanisms to determine the validity of a HOTP credential if supported. |
| | FIA_PSK_EXT.1/AuthSvr (selection-based) | Mitigates the threat by supporting PSKs in various forms as a form of claimant authenticator (password-based, HOTP, or TOTP). |
| | FIA_PSK_EXT.2 (selection-based) | Mitigates the threat by utilizing randomly generated PSKs as a form of claimant authenticator. |
| | FIA_PSK_EXT.3 (selection-based) | Mitigates the threat by utilizing password-based PSKs as a form of claimant authenticator and implementing mechanisms to determine the validity of a password-based PSK credential if supported. |
| | FIA_TOTP_EXT.1 (selection-based) | Mitigates the threat by utilizing TOTP as a form of claimant authenticator and implementing mechanisms to determine the validity of a TOTP credential if supported. |
| T.UNAUTHORIZED_ADMINISTRATOR_ACCESS (from NDcPP) | FAU_GEN.1/AuthSvr | Mitigates the threat by implementing an audit mechanism to detect potential misuse of the TOE. |

| | | |
|---|---|---|
| | FCS_CKM.3 | Mitigates the threat by implementing a cryptographic key access control mechanism to prevent compromise of the data in transit confidentiality mechanisms. |
| | FCS_STG_EXT.1 | Mitigates the threat by implementing a secure cryptographic key storage mechanism to prevent compromise of the data in transit confidentiality mechanisms. |
| | FIA_UAU.6 | Mitigates the threat by implementing a re-authentication mechanism to prevent compromise of an administrator account due to an unattended session. |
| | FMT_SMF.1/AuthSvr | Mitigates the threat by defining management functions as a legitimate mechanism to control the behavior of the TSF. |
| T.UNDETECTED_ACTIVITY (from NDcPP) | FAU_GEN.1/AuthSvr | Mitigates the threat by implementing an audit mechanism to detect potential misuse of the TOE. |
| | FCS_CKM.3 | Mitigates the threat by implementing a cryptographic key access control mechanism to prevent compromise of the data in transit confidentiality mechanisms. |
| | FCS_STG_EXT.1 | Mitigates the threat by implementing a secure cryptographic key storage mechanism to prevent compromise of the data in transit confidentiality mechanisms. |
| | FIA_UAU.6 | Mitigates the threat by implementing a re-authentication mechanism to prevent compromise of an administrator account due to an unattended session. |
| | FMT_SMF.1/AuthSvr | Mitigates the threat by defining management functions as a legitimate mechanism to control the behavior of the TSF. |

## 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 Device

### 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

**Table 4: Consistency of Security Problem Definition (NDcPP base)**

| 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 OE Objectives

**Table 5: Consistency of OE Objectives (NDcPP base)**

| 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 rationale for why this does not conflict with the claims defined by the NDcPP are as follows:

**Table 6: Consistency of Requirements (NDcPP base)**

| PP-Module Requirement | Consistency Rationale |
|---|---|
| **Modified SFRs** | |
| This PP-Module does not modify any requirements when the NDcPP is the base. | |
| **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. |
| 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-dependent SFRs** | |
| This PP-Module does not define any Implementation-dependent 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 or SARs.

## A.2 Objective Requirements

This PP-Module does not define any Objective SFRs.

## A.3 Implementation-dependent Requirements

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

# Appendix B - Selection-based Requirements

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

**Table 7: Auditable Events for Selection-based Requirements**

| Requirement | Auditable Events | Additional Audit Record Contents |
|---|---|---|
| FCS_RADSEC_EXT.1 | No events specified | N/A |
| FIA_HOTP_EXT.1 | Generation of a HOTP seed key. | Entity identifier. |
|  | Entity HOTP value comparison. | Result of comparison - success or failure. |
| FIA_PSK_EXT.1/AuthSvr | No events specified | N/A |
| FIA_PSK_EXT.2 | No events specified | N/A |
| FIA_PSK_EXT.3 | No events specified | N/A |
| FIA_TOTP_EXT.1 | Generation of a TOTP seed key. | Entity identifier. |
|  | Entity TOTP value comparison. | Result of comparison - success or failure. |

## B.2 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 **as defined in the Functional Package for X.509, version 1.0** 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 Functional Package for Transport Layer Security (TLS), version 2.1***, *DTLS in accordance with*

*FCS_DTLSS_EXT.1 and FCS_DTLSS_EXT.2* ***from the Functional Package for Transport Layer Security (TLS), version 2.1***] 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 Functional Package for Transport Layer Security (TLS), version 2.1" 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 Functional Package for Transport Layer Security (TLS), version 2.1" 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.

## B.3 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.1 of 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-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *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.

### 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.

### 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 256 bit-based pre-shared keys via FCS_RBG.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.

### 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-384, SHA-512]*], with [**assignment**: *positive integer of 4096 or more*] iterations, and output cryptographic key sizes [*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.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.

### 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.1 of 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-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *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.

# 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 8: Extended Component Definitions**

| Functional Class | Functional Components |
|---|---|
| Cryptographic Support (FCS) | FCS_EAPTLS_EXT EAP-TLS Protocol<br>FCS_RADIUS_EXT Authentication Protocol<br>FCS_RADSEC_EXT RadSec<br>FCS_STG_EXT Cryptographic Key Storage |
| Identification and Authentication (FIA) | FIA_HOTP_EXT HMAC-Based One-Time Password Pre-Shared Keys<br>FIA_PSK_EXT Pre-Shared Keys<br>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.1 Random Bit Generation<br>[FCS_TLSS_EXT.1 TLS Server Protocol Without Mutual Authentication, or<br>FCS_DTLSS_EXT.1 DTLS Server Protocol Without Mutual Authentication]<br>[FCS_TLSS_EXT.2 TLS Server Support for Mutual Authentication, or<br>FCS_DTLSS_EXT.2 DTLS Server Support for Mutual Authentication]<br>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.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, 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: FCS_CKM.3 Cryptographic Key Access

#### FCS_STG_EXT.1.1

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

- *encrypted with 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.1 Random Bit Generation
[FIA_PSK_EXT.1 Pre-Shared Key Composition, or
FIA_X509_EXT.1 X.509 Certificate Validation]

[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]

#### 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<br>FCS_RBG.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.1 of 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-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *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

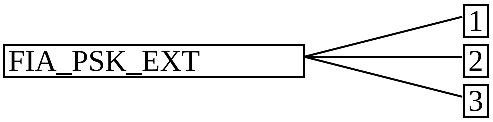
The TSF shall increment the counter after each successful authentication.

## 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<br>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 256 bit-based pre-shared keys via FCS_RBG.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<br>FCS_RBG.1 Random Bit Generation<br>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 [**assignment**: *key derivation algorithm*], with [**assignment**: *number or range of* *acceptable iterations*] iterations, and output cryptographic key sizes [**assignment**: *number of bits*] bits that meet the following: [**assignment**: *applicable standard*].

#### 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.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_NTP_EXT.1 NTP Protocol
FCS_RBG.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.1 of 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-384*, *SHA-512*] with key sizes [**assignment**: *key size (in bits) used in HMAC*] and message digest sizes [**selection**: *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, 8.3 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 (as defined in Functional Package for X.509, version 1.0) | X.509 Certificate Validation | 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

**Table 9: 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

**Table 10: Bibliography**

| Identifier | Title |
|---|---|
| [CC] | Common Criteria for Information Technology Security Evaluation -<br>• Part 1: Introduction and general model, CCMB-2022-11-001, CC:2022, Revision 1, November 2022.<br>• Part 2: Security functional requirements, CCMB-2022-11-002, CC:2022, Revision 1, November 2022.<br>• Part 3: Security assurance requirements, CCMB-2022-11-003, CC:2022, Revision 1, November 2022.<br>• Part 4: Framework for the specification of evaluation methods and activities, CCMB-2022-11-004, CC:2022, Revision 1, November 2022.<br>• Part 5: Pre-defined packages of security requirements, CCMB-2022-11-005, CC:2022, Revision 1, November 2022. |
| [CEM] | Common Methodology for Information Technology Security Evaluation -<br>• Evaluation methodology, CCMB-2022-11-006, CC:2022, Revision 1, November 2022. |
| [NDcPP] | collaborative Protection Profile for Network Devices, Version 4.0, December 22, 2025 |