Network Device Collaborative Protection Profile (NDcPP)/Application Software Protection Profile (App PP)
Extended Package (EP) for Authentication Servers

7 August 2015
Version 1.0

Table of Contents

1 Introduction

This Extended Package (EP) describes security requirements for an authentication server and is intended to provide a minimal baseline set of requirements that are targeted at mitigating well defined and described threats. Authentication server products allow enterprises to provide a centralized and standardized method of evaluating user authentication requests made throughout the enterprise. This enables centralized definition of 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. This EP focuses specifically on RADIUS authentication servers.

This introduction describes the features of a compliant Target of Evaluation (TOE) and discusses how this EP is to be used in conjunction with the Network Device collaborative Protection Profile (NDcPP) or the Application Software Protection Profile (App PP).

1.1 Conformance Claims

This EP serves to complement the NDcPP or the App PP with additional SFRs and associated Assurance Activities specific to the authentication server. Assurance Activities are the actions that the evaluator performs in order to determine a TOE's compliance to the SFRs.

This EP conforms to Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 3. It is CC Part 2 extended and CC Part 3 conformant.

1.2 How to Use This Extended Package

This EP extends the NDcPP when the authentication server is installed on a dedicated network appliance that is provided by the product vendor. This EP extends the App PP when the authentication server is a software application that is installed on a general purpose computer that is not provided by the product vendor.

As an EP of either the NDcPP or the App PP, it is expected that the content of this EP and the chosen base PP be appropriately combined in the context of each product-specific Security Target. This EP has been specifically defined such that there should be no difficulty or ambiguity in doing so. When this EP is used with the NDcPP or the App PP, conformant TOEs are obligated to implement the functionality required in those PPs with the additional functionality defined in this EP in response to the threat environment discussed subsequently herein. An ST must identify the applicable versions of the PP chosen and this EP in its conformance claims.

1.3 Compliant Targets of Evaluation

This document specifies Security Functional Requirements for an authentication server. An authentication server is designed to authenticate an entity (user or network device) that attempts to access a protected network. A Network Access Server (NAS) forwards authentication credentials to the authentication server; the authentication server verifies the credentials and determines whether the device or user is authorized. The authentication server defined by this EP could be either a stand-alone dedicated device or an application that runs on another system in the Operational Environment. The authentication server can also be co-located with the NAS, or separate.

An authentication server may be part of an AAA (authentication, authorization, and accounting) server. Authentication identifies the entity, authorization enforces access control policies, and accounting keeps

track of resources for billing and analysis purposes. This EP specifies the functional requirements for an AAA server's authentication services only.

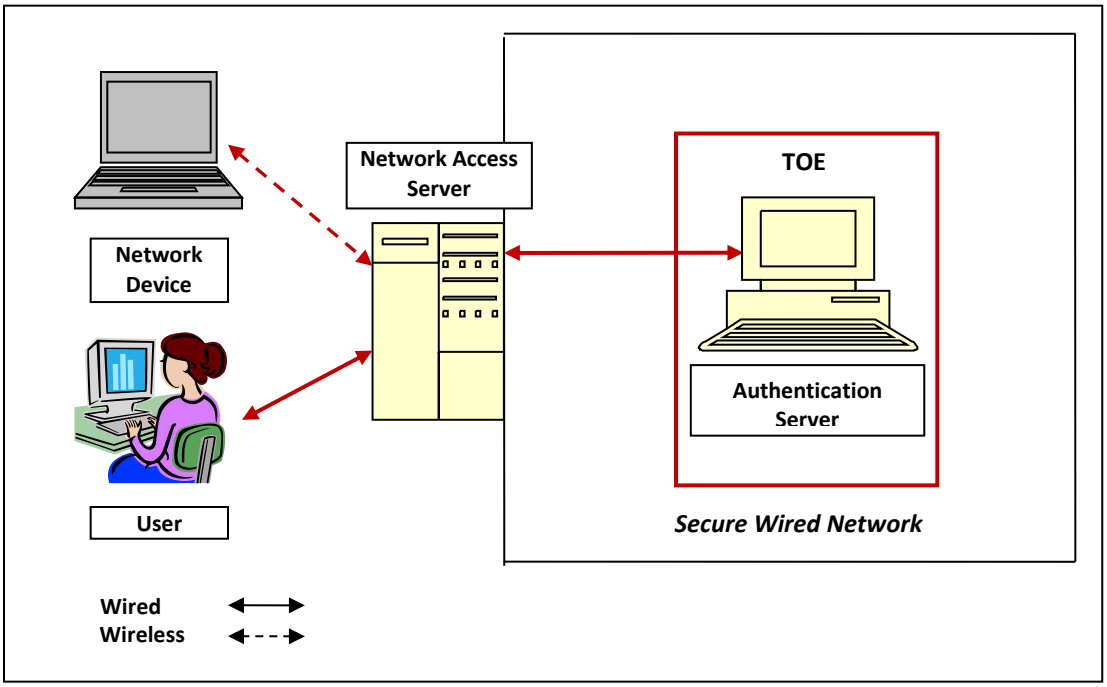

Figure 1: Network with an Authentication Server

1.4 Usage and Major Security Features of the TOE

Authentication servers are used to perform device and/or user authentication in support of a NAS. For example, a Wireless Local Area Network (WLAN) Access System may utilize the services of a dedicated authentication server during tunnel establishment. As a result, the authentication server must support IEEE 802.1X Port Based Network Access Control and must fulfill the IEEE 802.11 authentication server role. The architectural framework of Port-based access control defines three distinct roles: supplicant (client), authenticator (NAS); and authentication server (the TOE). The NAS requires the supplicant to perform 802.1X authentication, relying on the authentication server to authenticate the supplicant before providing network access. The NAS acts as a pass-through device between the supplicant and the authentication server.

Likewise, the authentication server may be used during Virtual Private Network (VPN) tunnel establishment. The VPN Gateway (NAS) acts as a pass through device between the VPN client and authentication server (the TOE). Regardless of connection type (or specific NAS), access is granted and secure communication tunnels are formed only if the authentication is successful.

The authentication server must support Remote Authentication Dial In User Service (RADIUS) in accordance with RFC 2865 for communication with the NAS. More specifically, RADIUS with Extensible Authentication Protocol – Transport Layer Security (EAP-TLS over RADIUS) as specified in RFC 5216, RFC 2865 and RFC 2869 is required for mutual authentication of the NAS and the TOE. All EAP-TLS communication between the TOE and the NAS is transported using the RADIUS protocol. In some scenarios, it may not be sufficient to protect authentication information solely with RADIUS. Therefore, additional security mechanisms must also be supported; the NAS and the TOE will be capable of

establishing a TLS connection using either RadSec or an IPSec tunnel between them to protect their communication link. This ensures that RADIUS attributes are transmitted securely and that authentication traffic remains confidential.

In addition to providing protected communications, the TOE must provide functionality for role separation and system monitoring. An administrative role is maintained such that only authenticated administrators are authorized access to the authentication server for installation, configuration, and maintenance. The authentication server supports both a remote authentication mechanism and a local authentication mechanism to perform administrator login. The administrator can access the TOE remotely through a secure connection implemented by IPSec, SSH, or TLS.

A robust audit mechanism is necessary to ensure accountability of all security relevant actions performed. The authentication server generates audit records and logs all critical authentication and system level events and all configuration options that were invoked by the authorized administrator. The audit records are protected against improper modification or deletion. The TOE restricts review of the audit log to the authorized administrator, providing administrators with the ability to detect and attribute malicious actions.

It is assumed that the authentication server is implemented properly and contains no critical design mistakes. The vendor is required to provide configuration guidance (AGD_PRE, AGD_OPE) to correctly install and administer the TOE for every operational environment supported.

1.4.1 Cryptography

The authentication server makes use of cryptographic algorithm implementations that are validated according to individual scheme policy.[1] All cryptographic functions are required to be approved, with implementation validated according to individual scheme policy, and running in an approved mode.[2]

1.4.2 Administration

The authentication server must provide an administrator role to install, configure, and maintain the TOE. The TOE will provide both remote and local authenticated access to perform administrative duties. The administrator can access the TOE remotely through a secure connection implemented by IPSec, SSH, or TLS. This interface may be provided by the TOE itself or, if the TOE is an application, it may be provided by the underlying platform on which the TOE resides. Although this EP requires one administrative role, the ST author can include additional administrative roles to further separate administrative functions into distinct administrative roles (e.g., cryptographic administrator, audit administrator). In this case, the ST author will need to refine the FMT_SMR requirement and update applicable security management requirements to restrict functionality to the appropriate administrator role.

Authorized administrators will correctly follow any required configuration guidance, as provided by the vendor for the evaluated configuration and dictated by the local organization. The TOE shall be capable of providing the following functions in addition to the functions defined by the claimed base PP:

- Implement the RADIUS protocol in accordance with applicable RFCs.
- Establish secure communications with a NAS.

[1] In the US scheme, the policy is to use FIPS certified modules and implementations.
[2] In the US scheme, all cryptographic functions are required to be FIPS approved, implemented in a FIPS validated module and running in a FIPS approved mode.

- Return appropriate authentication decisions to the NAS based on the contents of the received authentication requests and the subject data and authentication policies that are known to the TOE.
- Generate audit logs of security-relevant authentication server activity.
- Allow for administration of the behavior of the authentication server function.

Additionally, if the TOE claims conformance to the App PP as its base PP, the following functions are assumed to be provided by the Operational Environment but may be implemented by the TOE:

- Identify and authenticate administrators requesting the ability to manage the TOE.
- Securely transmit audit data to a remote repository.
- Provide a secure update mechanism.
- Provide cryptographic functions that are used to establish secure remote communications.
- Securely store administrative credential data.
- Securely store security-relevant cryptographic data.

The vendor is required to provide configuration guidance (AGD_PRE, AGD_OPE) to ensure correct installation and configuration of the TOE.

1.4.3 Protocol Compliance

The authentication server will support the RADIUS protocol (RFC 2865) and the RADIUS extension for EAP as specified in RFC 2869, for encapsulation of EAP packets transmitted between the TOE and the NAS. Support for transmitting the RADIUS EAP packets inside an IPSec VPN tunnel as specified in RFC 4301, or TLS encryption of RADIUS packets using RadSec as specified in RFC 6614 will be included. Pre-shared key authentication for this RadSec connection or IPSec tunnel is implemented; either RSA or ECDSA authentication must also be supported. The TOE must be configurable to disallow any authentication methods other than EAP-TLS. Allowed EAP-TLS cipher suites are specified in FCS_EAP-TLS_EXT.1; the authorized administrator configures the appropriate cipher suites depending on the connection type/NAS. Mutual authentication must occur; the authentication server uses an X.509 v3 machine certificate, generated by an approved CA, and public/private key pair to authenticate itself during the EAP-TLS exchanges. The authentication server authenticates a client entity using X.509 v3 certificates and validates the certificates in accordance with RFC 5280.

2 Security Problem Definition

The security problem faced for authentication servers is an extension of the threats that are defined for the base PPs. The SFRs that are taken from the base PPs or conditionally claimed based on the base PP chosen (see sections 4.3 and 4.4) address threats that are defined in the base PPs. However, the unique capabilities of the authentication server define the following additional threats:

2.1 False Endpoints

The TOE receives authentication requests from the environmental NAS component. The method by which the TSF signals acceptance or rejection of authentication requests must be transmitted using the RADIUS protocol in the manner specified by RFC 2865 so that this information can be interpreted by the NAS that resides in the Operational Environment. Additionally, the TOE and NAS must have mutual assurance of each other's identity. If they do not, the TSF could improperly return security-relevant authentication data to an entity impersonating the NAS or an attacker could impersonate the TSF and cause the NAS to perform improper actions in response to authentication requests.

T.FALSE_ENDPOINTS A malicious actor may falsely impersonate the TOE or the NAS in order to cause the TOE to operate in an insecure manner or to extract security-relevant data from the TOE or its Operational Environment.

2.2 Invalid Users

The primary purpose of the authentication server is to authenticate end users that are attempting to access protected resources in the Operational Environment. Therefore, it is essential that the TSF provide the ability to authenticate valid users and to ensure that invalid users cannot be authenticated. These operations can be applied to a variety of authentication scenarios.

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

The complete mapping of the security problem is found in Appendix A of this EP.

3 Security Objectives

The security problem described in Section 2 will be addressed by the mandatory SFRs that are defined for this EP. SFRs that are taken from either of the base PPs will be used to address threats that are defined in those PPs. If the TOE claims conformance to the base App PP, it will also satisfy the threats defined for the base NDcPP in addition to those defined specifically for the App PP either through its own functionality or through security that is assumed to be provided by the underlying platform (see section 4.4.2 for guidance on this). If the TOE claims conformance to the base NDcPP, the TOE boundary includes the underlying platform so the App PP requirements, which primarily focus around the interaction of the application with its environmental platform, are not expected to apply.

Note: in each subsection below particular security objectives are identified (highlighted by O.) and they are matched with the associated SFRs that provide the mechanisms to satisfy the objectives.

3.1 Standardized Protocol Usage

To ensure that the TOE can properly authenticate valid users, the TSF must implement the RADIUS protocol according to specification. This includes not only the proper handling of authentication requests but also verification of the correctness and integrity of the packets in which the requests are included. Failure to do so will result in a lack of assurance of its proper functionality.

(O.RADIUSCOMPLIANT -> FCO_NRO.1, FCO_NRR.1, FCS_RADIUS_EXT.1, FCS_RADSEC_EXT.1)

3.2 Mutual Authentication

To reduce the risk of impersonation of either the TOE or the NAS that it receives authentication requests from, the TOE is expected to implement measures that allow for it to identify and authenticate the NAS, and for it to provide information to the NAS that will allow for mutual authentication.

(O.TRUSTEDNAS -> FCS_EAP-TLS_EXT.1, FIA_PSK_EXT.1, FTP_ITC.1)

3.3 Authoritative User Authentication

The TOE must be able to make authentication decisions based on the identity and credential data that it receives. The TOE may return different authentication results based on contextual data such as the day/time of the attempt, the type of credential being used, or other attributes that are assigned to the subject. This attribute data may be maintained by the TSF or the TOE may provide the ability to communicate with a repository in the Operational Environment where this data is maintained.

(O.USERAUTH -> FIA_AFL.1, FIA_UAU.6, FTA_TSE.1)

4 Security Functional Requirements

This section specifies a Security Functional Requirement for the TOE, as well as specifying the assurance activities the evaluator performs.

4.1 Conventions

While the SFR in this EP is extended, it is defined in a flexible manner for use in this and other EPs, or PPs, and as such operations are performed in the context of this EP.

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text and beginning with "assignment:";
- Refinement made by EP author: Indicated with **bold text** (added text) and ~~strikethroughs~~ (removed text), if necessary;
- Selection: Indicated with underlined text and beginning with "selection:";
- Assignment within a Selection: Indicated with *italicized and underlined* text; and
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).

All selection and assignment operations are further denoted by the use of square brackets. In some cases, the EP requires that selection and/or assignment operations be completed using specific wording. When this occurs, the formatting for the operation is preserved except for the initial "selection:" or "assignment:" wording, which is removed.

4.2 EP Security Functional Requirement Direction

The following section describes the SFRs that must be satisfied by any TOE that claims conformance to this EP. These SFRs must be claimed regardless of whether the base PP is the NDcPP or the App PP.

Table 4-1: Security Functional Requirements

Class Name	Component Identification	Component Name
FCO: Communications	FCO_NRO.1	Selective Proof of Origin
	FCO_NRR.1	Selective Proof of Receipt
FCS: Cryptographic Support	FCS_EAP-TLS_EXT.1	Extended: Extensible Authentication Protocol – Transport Layer Security
	FCS_RADIUS_EXT.1	Extended: RADIUS
	FCS_RADSEC_EXT.1	Extended: RadSec
FIA: Identification and Authentication	FIA_PSK_EXT.1	Extended: Pre-Shared Key Composition
FTA: TOE Access	FTA_TSE.1	TOE Session Establishment
FTP: Trusted Path/Channels	FTP_ITC.1	Inter-TSF Trusted Channel

4.2.1 Communications (FCO)

4.2.1.1 FCO_NRO.1 Selective Proof of Origin

FCO_NRO.1.1 The TSF shall be able to generate evidence of origin for transmitted [*RADIUS Access-Request packets*] at the request of the [recipient].

Application Note: *The intent of this requirement is for the TOE to have the ability to validate the NAS and prevent this component from being spoofed. In this case, the TSF is the recipient of the transmitted Access-Request and it must have the ability to identify where it was received from.*

FCO_NRO.1.2 The TSF shall be able to relate the [*Message Authenticator*] of the originator of the information, and the [*Access-Request*] of the information to which the evidence applies.

Application Note: *The intent of this requirement is for the TOE to be able to validate the authenticity of the NAS specifically by verifying the Message Authenticator that is computed in part using a shared secret known to both the NAS and the TOE. This behavior is defined in RFC 3579.*

FCO_NRO.1.3 The TSF shall provide a capability to verify the evidence of origin of information to [recipient] given [*the evidence of origin information is presented in the Access-Request packet in a manner consistent with RFC 2865*].

Assurance Activity	
TSS	The evaluator shall check the description of the implementation of this protocol to ensure that RADIUS encapsulated EAP Message Authenticators conform to RFC 3579.
AGD	The evaluator shall verify that the guidance contains all necessary instructions to configure RADIUS and encapsulated EAP on the TOE, in order to ensure that evidence of origin for all incoming RADIUS Access-Request packets is collected and preserved.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall send a RADIUS Access-Request, from a NAS with which the TOE does not share a RADIUS secret, with NAS identification attributes correctly indicating the originating NAS, containing an encapsulated EAP-response message and a valid message-authenticator attribute. The evaluator shall verify that the TOE discards the request without responding. Test 2: The evaluator shall send a RADIUS Access-Request, from a NAS with which the TOE does not share a RADIUS secret, with NAS identification attributes falsely indicating a NAS with which the TOE does share a RADIUS secret, containing an encapsulated EAP-response message and a valid message-authenticator attribute. The evaluator shall verify that the TOE discards the request without responding.

4.2.1.2 FCO_NRR.1 Selective Proof of Receipt

FCO_NRR.1.1 The TSF shall be able to generate evidence of receipt for received [*RADIUS Access-Request packets*] 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 NAS upon receipt of an Access-Request.*

FCO_NRR.1.2 The TSF shall be able to relate the [*Identifier, Response Authenticator*] of the recipient of the information, and the [*response packet*] 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 the response packet (Access-Accept, Access-Request, or Access-Challenge) that identifies:*

- *the Access-request that is being responded to;*
- *the Response Authenticator that identifies the TOE as the valid recipient of the original Access-Request, based in part on the shared secret known to both the NAS and the TOE.*

This behavior is defined in RFC 2865.

FCO_NRR.1.3 The TSF shall provide a capability to verify the evidence of receipt of information to [originator] given [*assignment: limitations on the evidence of receipt*].

Assurance Activity	
TSS	The evaluator shall check the description of the implementation of this protocol to ensure that RADIUS Response Authenticators conform to RFC 2865.
AGD	The evaluator shall verify that the guidance contains all necessary instructions to configure RADIUS and

	encapsulated EAP on the TOE, in order to ensure that evidence of receipt of all incoming RADIUS Access-Request packets is generated and transmitted correctly.
Test	The evaluator shall perform the following test: Test 1: The evaluator shall send a RADIUS Access-Request containing an encapsulated EAP-response message of type Identity, specifying a valid user account, a service for which the user is authorized, and containing all information required to authenticate the user. The evaluator shall verify that the TOE returns an Access-Challenge, and that the MD5 hash of the concatenated Code + ID + Length + Request Authenticator of the Access-Request + Attributes + Secret matches the response authenticator.

4.2.2 Cryptographic Support (FCS)

4.2.2.1 FCS_EAP-TLS_EXT.1 Extended: Extensible Authentication Protocol – Transport Layer Security (EAP-TLS)

FCS_EAP-TLS_EXT.1.1 The TSF shall implement **EAP-TLS protocol as specified in RFC 5216 with** [selection: TLS 1.1 (RFC 4346), TLS 1.2 (RFC 5246)], and no other TLS version, and support the following ciphersuites:

Mandatory Ciphersuites in accordance with RFC 3268:

- TLS_RSA_WITH_AES_128_CBC_SHA

Optional Ciphersuites:

[selection:
- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5246
- TLS_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_128_CBC_ SHA256 as defined in RFC 5246
- TLS_DHE_RSA_WITH_AES_256_CBC_ SHA256 as defined in RFC 5246
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 as defined in RFC 5289
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256 as defined in RFC 5430
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384 as defined in RFC 5430
- no other ciphersuite

].

Application Note: The ciphersuites to be tested in the evaluated configuration are limited by this requirement. The ST author should select the optional ciphersuites that are supported; if there are no ciphersuites supported other than the mandatory suites, then “None” should be selected. It is necessary to limit the ciphersuites that can be used in an evaluated configuration administratively on the server in the test environment. The Suite B algorithms listed above (RFC 6460) are the preferred algorithms for implementation. TLS_RSA_WITH_AES_128_CBC_SHA is required in order to ensure compliance with RFC 5246.

If an elliptic-curve ciphersuite is selected, FCS_EAP-TLS_EXT.1.7 in Appendix C shall be included in the ST.

TLS 1.2 is the preferred protocol. TLS 1.1 does not have the extensions necessary to assure a connection with security strength of 112-bits or better.

TLS 1.2 is required for EAP-TLS for products entering into evaluation after Quarter 3, 2015. These requirements will be revisited as new TLS versions are standardized by the IETF.

FCS_EAP-TLS_EXT.1.2 The TOE shall generate random values used in the EAP-TLS exchange using the RBG specified in FCS_RBG_EXT.1

FCS_EAP-TLS_EXT.1.3 The TSF shall request a certificate from the client, requiring client authentication.

FCS_EAP-TLS_EXT.1.4 The TSF shall verify that the client certificate presented includes the Client Authentication purpose (OID 1.3.6.1.5.5.7.3.2) in the extended KeyUsage field and the Key Agreement bit is set in the KeyUsage field (OID 2.5.29.15.4).

FCS_EAP-TLS_EXT.1.5 The TSF shall use X509 v3 certificates as specified in FIA_X509_EXT.1.

FCS_EAP-TLS_EXT.1.6 The TSF shall allow an authorized administrator to configure the list of algorithm suites that may be proposed and accepted during the EAP-TLS exchanges.

Application Note: FCS_EAP-TLS_EXT.1.4 requires that the key agreement bit be set in support of the mandatory Diffe-Helman ciphersuites. If the ST author chooses to use an optional RSA ciphersuite, this component should be iterated and key agreement bit with the "key agreement bit" replaced with "key encipherment bit".

Assurance Activity	
TSS	The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that optional characteristics (e.g., extensions supported) are specified as well as the supported ciphersuites. The evaluator shall check the TSS to ensure that the ciphersuites specified are identical to those listed for this component.
AGD	The evaluator shall check that the operational guidance contains instructions for the administrator to configure the list of Certificate Authorities that are allowed to sign certificates used by the authentication server that will be accepted by the TOE in the EAP-TLS exchange, and instructions on how to specify the algorithm suites that will be proposed and accepted by the TOE during the EAP-TLS exchange. The evaluator shall also check the operational guidance to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements).
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an EAP session. It is sufficient to observe (on the wire) the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES). Test 2: The following test is repeated for each supported certificate signing algorithm supported. The evaluator shall attempt to establish the connection such that the client certificate contains the Client Authentication purpose in the extended KeyUsage field and the Key Agreement bit is set in the KeyUsage field and verify that a connection is established. The evaluator will then verify that connection is not

	established with an otherwise valid client certificate that lacks the Client Authentication purpose in the extended KeyUsage field. Test 3: The evaluator shall follow the administrative guidance to configure the list of protocols to be proposed during EAP-TLS negotiations that is limited to only those specified by the first element of this component. The evaluator shall initiate a connection with a NAS and ensure that only those protocols configured are proposed. If the initial list is not a subset of the total set of protocol proposed by the client, the evaluator shall repeat the test specifying a subset of the protocols used in the initial test.

4.2.2.2 FCS_RADIUS_EXT.1 Extended: RADIUS

FCS_RADIUS_EXT.1.1 The TSF shall implement the RADIUS protocol as specified in RFC 2865 for communication with a NAS.

FCS_RADIUS_EXT.1.2 The TSF shall implement RADIUS encapsulated EAP, as specified in RFC 3579.

FCS_RADIUS_EXT.1.3 The RADIUS extension for EAP (RFC 2869) shall support the use of EAP-TLS for authentication as specified in RFC 5216.

Application Note: *The ST author should describe how the TSF determines the situations under which issuing an Access-Challenge response to the NAS is necessary, apart from those required by the EAP-TLS protocol.*

Assurance Activity	
TSS	The evaluator shall examine the TSS to ensure that RADIUS is specified as the protocol by which all communication between the TOE and the NAS is conducted. The evaluator shall examine the TSS to ensure that EAP is specified as the authentication protocol to be used between the TOE and the NAS, that TLS is the means of mutual authentication to be carried out over EAP, and that other authentication frameworks are disallowed. The evaluator shall check the description of the implementation of this protocol to ensure that RADIUS encapsulated EAP Message Authenticators conform to RFC 3579.
AGD	The evaluator shall verify that the guidance contains all necessary instructions to configure RADIUS and encapsulated EAP-TLS on the TOE, in accordance with RFCs 2865, 2869, 3579, and 5216.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall send RADIUS access-requests with encapsulated EAP-response messages to the TOE, from a NAS with which the TOE shares a RADIUS pre-shared key, and verify that the TOE responds appropriately according to RFCs 2865 and 3579: • An access-request containing an encapsulated EAP-request message. The evaluator shall verify that the TOE returns an access-reject containing an encapsulated EAP-response of type Nak, indicating no alternatives. • Access-requests containing encapsulated EAP-response messages and each of the following attributes: User-password, CHAP-password, CHAP-challenge, ARAP-password, password-retry, reply-message, error-cause. The evaluator shall verify that in each case, the TOE discards the request without responding. • An access-request containing an encapsulated EAP-response message, but no message-authenticator attribute. The evaluator shall verify that the TOE discards the request without responding. • An access-request containing an encapsulated EAP-response message of type MD5-challenge. The evaluator shall verify that the TOE responds with an access-challenge message of type Nak or expanded Nak.

	• An access-request containing an encapsulated EAP-response message of type Identity, specifying a valid user account, a service for which the user is authorized, and containing all information required to authenticate the user.
	  ○ The evaluator shall verify that the TOE returns an access-challenge with an encapsulated EAP-TLS start packet; i.e. an EAP-request with EAP-type set to EAP-TLS, the start bit set, and no data.
	  ○ The evaluator shall go on to complete the TLS handshake, presenting valid, untrusted, expired, and revoked client certificates to the TOE, and verify that the handshake completes successfully only for valid certificates, and unsuccessfully otherwise,
	  ○ The evaluator shall verify that the TOE indicates a successful TLS handshake with an access-accept with encapsulated EAP-success packet. The evaluator shall verify that the TOE indicates an unsuccessful TLS handshake with an access-reject with encapsulated EAP-failure packet.
	  ○ During an otherwise successful handshake, the evaluator shall send an access-request with encapsulated EAP-response with EAP-type set to anything but EAP-TLS, and verify that the TOE returns an access-challenge with encapsulated EAP-request of type EAP-TLS, indicating error-cause: invalid EAP type error (ignored). The evaluator shall verify that subsequent handshake steps complete normally.
	  ○ During an otherwise successful handshake, the evaluator shall send five access-requests with encapsulated invalid EAP packets, and verify that the TOE returns an access-reject with encapsulated EAP-failure packet after receiving the fifth invalid packet.
	• An access-request containing an encapsulated EAP-response message of type Identity, specifying a valid user account, and a service for which the user is not authorized. The evaluator shall verify that the TOE returns an access-reject.
	• An access-request containing an encapsulated EAP-response message of type Identity, specifying an invalid user account. The evaluator shall verify that the TOE returns an access-reject.
	• An Access-Request whose length field is incorrect. The evaluator shall verify that the TOE discards the request without responding.
	• An Access-Request whose code field is invalid. The evaluator shall verify that the TOE discards the request without responding.
	• An Access-Request containing an encapsulated EAP-response message and a message-authenticator attribute that does not match the request. The evaluator shall verify that the TOE discards the request without responding.

4.2.3 Identification and Authentication (FIA)

4.2.3.1 FIA_PSK_EXT.1 Extended: Pre-Shared Key Composition

FIA_PSK_EXT.1.1 The TSF shall be able to use pre-shared keys for **[selection: IPsec, RadSec]** and [*RADIUS*].

Application Note: *The selection of IPsec or RadSec must match the selection in FTP_ITC.1. The pre-shared key used for RADIUS is the RADIUS shared secret.*

FIA_PSK_EXT.1.2 The TSF shall be able to accept text-based pre-shared keys that:

- are 22 characters and [selection: [*assignment: other supported lengths*], no other lengths];
- are composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")").

Assurance Activity	
TSS	The evaluator shall examine the TSS to ensure that it identifies all protocols that allow text-based pre-shared keys and states that text-based pre-shared keys of 22 characters are supported. For each protocol

	identified by the requirement. The evaluator shall also verify that the selection of IPsec or RadSec matches the selection in FTP_ITC.1.
AGD	The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on the composition of strong text-based pre-shared keys and (if the selection indicates keys of various lengths can be entered) that it provides information on the range of lengths supported. The guidance must specify the allowable characters for pre-shared keys and that list must be a super-set of the list contained in FIA_PSK_EXT.1.2.
Test	The evaluator shall also perform the following tests for each protocol (or instantiation of a protocol, if performed by a different implementation on the TOE). Note that one or more of these tests can be performed with a single test case. Test 1: The evaluator shall compose a pre-shared key of 22 characters that contains a combination of the allowed characters in accordance with the operational guidance, and demonstrates that a successful protocol negotiation can be performed with the key. Test 2 [conditional]: If the TOE supports pre-shared keys of multiple lengths, the evaluator shall repeat Test 1 using the minimum length; the maximum length; a length inside the allowable range; and invalid lengths beyond the supported range (both higher and lower). The minimum, maximum, and included length tests should be successful, and the invalid lengths must be rejected by the TOE.

4.2.4 TOE Access (FTA)

4.2.4.1 FTA_TSE.1 TOE Session Establishment

FTA_TSE.1.1 Refinement: The TSF shall be able to deny **user** session establishment based on [*assignment: attributes*].

Application Note: *The ST author should describe any circumstances that would cause a user's authentication request to be rejected. All compliant TOEs will reject authentication requests based on invalid credentials but some may impose additional limitations such as suspended user accounts or time of day restrictions, depending on the capabilities of the TSF's authentication mechanism.*

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that all of the attributes on which a user session can be denied are specifically defined.
AGD	The evaluator shall examine the operational guidance to determine that it contains guidance for configuring each of the attributes identified in the TSS.
Test	The evaluator shall also perform the following test for each attribute: Test 1: The evaluator shall successfully establish a user session. The evaluator shall follow the operational guidance to configure the system so that that user's access is denied based on a specific value of an attribute. The evaluator shall then attempt to establish a session in contravention to the attribute setting while still providing valid authentication data. The evaluator shall observe that the access attempt fails. The evaluator shall repeat this test for each attribute indicated by the ST author.

4.2.5 Trusted Path/Channels (FTP)

4.2.5.1 FTP_ITC.1 Inter-TSF Trusted Channel

FTP_ITC.1.1 Refinement: The TSF shall provide **[selection: an IPsec, a RadSec]** communication channel between itself and ~~another trusted IT product~~ **a NAS** 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 of the channel data.

FTP_ITC.1.2 Refinement: The TSF shall permit [the TSF, **or the NAS**] to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate the communication via the trusted channel for [*responses to RADIUS Access-Request messages received from the NAS*].

Application Note: If this EP is used to extend the NDcPP, FTP_ITC.1 will already exist in the base PP. In this case the ST author is expected to apply an iteration operation to both instances of FTP_ITC.1 in order for the SFR names to remain unique. The evaluator is also expected to perform the assurance activities for each iteration of the SFR in this case.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that, for all communications with authorized IT entities identified in the requirement, each communications mechanism is identified in terms of the allowed protocols for that IT entity. The evaluator shall also confirm that all protocols listed in the TSS are specified and included in the requirements in the ST.
AGD	The evaluator shall confirm that the guidance documentation contains instructions for establishing and re-establishing the allowed protocols with each authorized IT entity.
Test	The evaluator shall perform the following tests: Test 1: The evaluators shall ensure that communications using each protocol with each NAS is tested during the course of the evaluation, setting up the connections as described in the guidance documentation and ensuring that communication is successful. Test 2: For each protocol that the TOE can initiate as defined in the requirement, the evaluator shall follow the guidance documentation to ensure that in fact the communication channel can be initiated from the TOE. Test 3: The evaluator shall ensure, for each communication channel with a NAS, the channel data uses the appropriate identified protocols. Test 4: The evaluators shall, for each protocol associated with each NAS tested during test 1, physically interrupt an established connection. The evaluator shall ensure that when physical connectivity is restored, communications are appropriately protected. Further assurance activities are associated with the specific protocols based on the selection that is chosen in FTP_ITC.1.1.

4.3 NDcPP Security Functional Requirement Direction

If this EP is extending the NDcPP, the authentication server is expected to rely on the security functions implemented by the network device as a whole and evaluated against the base PP. If a TOE claiming conformance to this EP is using the NDcPP as the claimed base PP, the following sections describe any modifications that the ST author must make to the SFRs defined in the base PP in addition to what is mandated by section 4.2 above.

4.3.1 Inclusion of Optional Requirements

In order for the TOE to satisfy all necessary dependencies, the NDcPP claim must include all of the base requirements from that PP as well as any selection-based requirements pertaining to protocols used by

the TOE. If other optional SFRs from the NDcPP are applicable to the TOE, they should be claimed but are not mandated by this EP.

4.3.2 Security Audit (FAU)

4.3.2.1 FAU_GEN.1 Audit Data Generation

Application Note: *There are no additional SFRs for security audit. However, there are additional auditable events that serve to extend the FAU_GEN.1 SFR found in the NDcPP. As such, the following events should be combined with those of the NDcPP in the context of a conforming Security Target.*

Table 4-2: Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FCO_NRO.1	Client request for which the TOE does not have a shared secret	Identity of the client, contents of EAP-response (if present).
FCO_NRR.1	None	None
FCS_EAP-TLS_EXT.1	Protocol failures Establishment of a TLS session	If failure occurs, record a descriptive reason for the failure
FCS_RADIUS_EXT.1	Protocol failures Success/Failure of authentication	If failure occurs, record a descriptive reason for the failure
FCS_RADSEC_EXT.1 (selection-based)	Failure to establish RadSec session	Reason for failure
FIA_AFL.1	The reaching of the threshold for the unsuccessful authentication attempts. Disabling an account due to the threshold being reached	The claimed identity of the user attempting to gain access or the IP where the attempts originated.
FIA_PSK_EXT.1	None	None
FIA_UAU.6 (optional)	All use of the authentication mechanism	Origin of the attempt (e.g., IP address)
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism	Reason for denial, origin of establishment attempt.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions	Identification of the initiator and target of failed trusted channels establishment attempt

Assurance Activity
The evaluator shall complete the assurance activity for FAU_GEN.1 as described in the NDcPP for the auditable events defined above in addition to the applicable auditable events that are defined in the NDcPP. The evaluator shall also ensure that the administrative actions defined for this EP are appropriately audited.

4.3.3 Identification and Authentication (FIA)

4.3.3.1 FIA_AFL.1 Authentication Failure Handling

FIA_AFL.1.1 Refinement: The TSF shall detect when **an Administrator configurable positive integer of successive** unsuccessful authentication attempts occur related to **administrators attempting to authenticate remotely**.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been **met**, the TSF shall [selection, choose one of: prevent the offending remote administrator from successfully authenticating until [assignment: *action*] is taken by a local Administrator; prevent the offending remote administrator from successfully authenticating until an Administrator defined time period has elapsed].

Application Note: This requirement does not apply to an administrator at the local console, since it does not make sense to lock a local administrator's account in this fashion. This could be addressed by (for example) requiring a separate account for local administrators or having the authentication mechanism implementation distinguish local and remote login attempts. The "action" taken by a local administrator is implementation specific and would be defined in the administrator 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.

Assurance Activity	
TSS	The evaluator shall examine the TSS to determine that it contains a description, for each supported method for remote administrative actions, of how successive unsuccessful authentication attempts are detected and tracked. The TSS shall also describe the method by which the remote administrator is prevented from successfully logging on to the TOE, and the actions necessary to restore this ability.
AGD	The evaluator shall also examine the operational guidance to ensure that instructions for configuring the authentication failure threshold and the TOE's response to the threshold being met (if configurable), and that the process of allowing the remote administrator to once again successfully log on is described for each "action" specified (if that option is chosen). If different actions or mechanisms are implemented depending on the trusted path used to access the TSF (see FTP_TRP.1), all must be described.
Test	The evaluator shall perform the following tests for each method by which remote administrators access the TOE: Test 1: The evaluator shall use the operational guidance to configure the number of successive unsuccessful authentication attempts allowed by the TOE. The evaluator shall test that once the limit is reached for a given remote administrator account, subsequent attempts with valid credentials are not successful. Test 2: [conditional] If the TSS indicates that administrative action is necessary to re-enable an account that was locked out due to excessive authentication failures, the evaluator shall perform the steps in Test 1 to lock out an account, follow the operational guidance to manually re-enable the locked out administrator account, and observe that it is once again able to successfully log in. Test 3: [conditional] If the TSS indicates that an administrator-configurable time period must elapse in order to automatically re-enable an account that was locked out due to excessive authentication failures, the evaluator shall perform the steps in Test 1 to lock out an account, follow the operational guidance to configure a time period of their choosing, and observe through periodic login attempts that the account cannot successfully log in until the configured amount of time has elapsed. The evaluator shall then repeat this test for a different time period of their choosing.

4.3.4 Security Management (FMT)

4.3.4.1 FMT_SMF.1 Specification of Management Functions

Application Note: *There are no additional SFRs for security management. However, there are additional management functions that serve to extend the FMT_SMF.1 SFR found in the NDcPP. As such, the following events should be combined with those of the NDcPP in the context of a conforming Security Target.*

- **Ability to configure the RADIUS shared secret**
- **Ability to define an authorized NAS**
- **Ability to enable, disable, and determine and modify the behavior of all the security functions of the TOE identified in this EP to the administrator**
- **[selection: Ability to configure the IPsec functionality,**
- **No other functions]**

Assurance Activity
Compliance with the other SFRs in section 4.2 and 4.3 of this EP as well as any applicable SFRs in the appendices is sufficient to demonstrate that the TOE provides sufficient means to manage its authentication server functions.

4.4 App PP Security Functional Requirement Direction

If this EP is extending the App PP, the authentication server is expected to rely on the security functions that are generic to all applications (and/or those that are provided by the underlying platform on which the application resides) and evaluated against the base PP. If a TOE claiming conformance to this EP is using the App PP as the claimed base PP, the following sections describe any modifications that the ST author must make to the SFRs defined in the base PP in addition to what is mandated by section 4.2 above.

4.4.1 Inclusion of Optional Requirements

In order for the TOE to satisfy all necessary dependencies, the App PP claim must include all of the base requirements as well as the following selection-based SFRs:

- FCS_CKM_EXT.1
- FIA_X509_EXT.1

It is necessary to include these SFRs, even if the pertinent selections are not made in the base App PP SFRs, because the specific functionality required by this EP has these functions as dependencies. If other optional, selection-based, or objective SFRs from the App PP are applicable to the TOE, they should be claimed but are not mandated by this EP.

4.4.2 Capabilities of the Underlying Platform

As an extension of the App PP, a TOE claiming conformance to this EP is assumed to rely on its underlying platform for a number of security functions that are defined in the NDcPP because they are not functions that are universally provided by software applications. However, it may be the case that the TOE provides these capabilities itself rather than relying fully on its Operational Environment. In these cases, the ST author is expected to claim relevant SFRs from the NDcPP that describe how the functionality is performed. The following table lists the relevant cases and the SFRs that should be claimed to address them. **Note that unless specified otherwise, the SFR definition and corresponding assurance activities should be taken directly from the NDcPP.**

Table 4-3: Security Functional Requirements

Capability Provided by the TOE	SFRs to Include
A distinct authentication mechanism that is not inherited from the underlying platform.	FIA_AFL.1 (see section 4.3.3.1)
	FIA_PMG_EXT.1
	FIA_UIA_EXT.1
	FIA_UAU_EXT.2
	FIA_UAU.7
	FTA_TAB.1
The ability to transmit audit data to a remote audit repository using a logging mechanism that is not provided by the underlying platform.	FAU_STG_EXT.1
The ability to update itself independent of a package manager or other capability provided by the underlying platform.	FMT_MOF.1(1)/TrustedUpdate
	FPT_TUD_EXT.1
The ability to use its own internal cryptographic module to establish an IPsec connection to the NAS.	FCS_IPSEC_EXT.1

The responsibility for storage of administrative credentials independent of (or in addition to) the protections provided by the underlying platform.	FPT_APW_EXT.1
The responsibility for storage of cryptographic security data independent of (or in addition to) the protections provided by the underlying platform.	FPT_SKP_EXT.1

4.4.3 Security Audit (FAU)

4.4.3.1 FAU_GEN.1 Audit Data Generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shut-down of the audit functions
b) All auditable events for the [not specified] level of audit; **and**
c) **All administrative actions comprising:**
    1. **Configuration of the RADIUS shared secret**
    2. **Configuration changes relating to communications with the NAS or other components residing in the Operational Environment**
    3. **Configuration of cryptographic or security-relevant settings**
    4. **Configuration of policies, schemes, or other factors controlling how the TSF enforces user authentication**
d) **Specifically defined auditable events listed in Table 4-4.**

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
b) For each audit event type, based on the auditable event definitions of the functional components included in the EP/ST**, information specified in column three of Table 4-4.**

Application Note: *The following table defines the auditable events for the SFRs that are mandated by the EP in section 4.2 as well as those mandated specifically for application TOEs in section 4.4. If the TOE claims any SFRs beyond this, the ST author shall add to this table all relevant auditable events for the SFRs that are claimed from their proper source, e.g. any optional SFRs that are included from the NDcPP as per section 4.4.2 shall also include the corresponding auditable events defined in the NDcPP.*

Table 4-4: Auditable Events

Requirement	Auditable Events	Additional Audit Record Contents
FCO_NRO.1	Client request for which the TOE does not have a shared secret	Identity of the client, contents of EAP-response (if present).
FCO_NRR.1	None	None
FCS_EAP-TLS_EXT.1	Protocol failures Establishment of a TLS session	If failure occurs, record a descriptive reason for the failure
FCS_RADIUS_EXT.1	Protocol failures Success/Failure of authentication	If failure occurs, record a descriptive reason for the failure
FCS_RADSEC_EXT.1 (selection-based)	Failure to establish RadSec session	Reason for failure
FIA_PSK_EXT.1	None	None
FIA_UAU.6 (optional)	All use of the authentication mechanism	Origin of the attempt (e.g., IP address)
FMT_SMR.2	a) Modifications to the group of users that are part of a role; b) Unsuccessful attempts to use a role due to given	User IDs which are associated with the modifications

	conditions on the roles	The identity of the administrator performing the function
FPT_TST_EXT.1	Execution of this set of cryptographic module self-tests	No additional information
FTA_TSE.1	Denial of a session establishment due to the session establishment mechanism.	Reason for denial, origin of establishment attempt.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt.

Assurance Activity	
TSS	The evaluator shall check the TSS and ensure that it provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field.
AGD	The evaluator shall check the operational guidance and ensure that it lists all of the auditable events and provides a format for audit records. Each audit record format type must be covered, along with a brief description of each field. The evaluator shall check to make sure that every audit event type mandated by the EP is described and that the description of the fields contains the information required in FAU_GEN.1.2, and the additional information specified in Table 4-4. The evaluator shall in particular ensure that the operational guidance is clear in relation to the contents for failed events. In Table 4-4, information detailing the event and a name or identifier for the originator of the event is required. The evaluator shall ensure that name or identifier is sufficient to allow an administrator reviewing the audit log to determine the context of the operation as well as the non-TOE endpoint of the connection for failures relating to communications with other IT systems. The evaluator shall also make a determination of the administrative actions that are relevant in the context of this EP. The TOE may contain functionality that is not evaluated in the context of this EP because the functionality is not specified in an SFR. This functionality may have administrative aspects that are described in the operational guidance. Since such administrative actions will not be performed in an evaluated configuration of the TOE, the evaluator shall examine the operational guidance and make a determination of which administrative commands, including subcommands, scripts, and configuration files, are related to the configuration (including enabling or disabling) of the mechanisms implemented in the TOE that are necessary to enforce the requirements specified in the EP, which thus form the set of "all administrative actions". The evaluator may perform this activity as part of the activities associated with ensuring the AGD_OPE guidance satisfies the requirements.
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall test the TOE's ability to correctly generate audit records by having the TOE generate audit records in accordance with the assurance activities associated with the functional requirements in this EP. Test 2: The evaluator shall test that each administrative action applicable in the context of this EP is auditable. When verifying the test results, the evaluator shall ensure the audit records generated during testing match the format specified in the administrative guide, and that the fields in each audit record have the proper entries. Note: Testing here can be accomplished in conjunction with the testing of the security mechanisms directly. For example, testing performed to demonstrate that FTE_TSE.1 appropriately performs denial of session establishment can also be used to demonstrate that an audit record of this function was appropriately generated. Separate tests that exclusively test the audit functionality are not required if the evaluator can identify other tests in which each auditable event was generated.

4.4.4 Security Management (FMT)

4.4.4.1 FMT_SMF.1 Specification of Management Functions

Application Note: *There are no additional SFRs for security management. However, there are additional management functions that serve to extend the FMT_SMF.1 SFR found in the App PP. As such, the following events should be combined with those of the App PP in the context of a conforming Security Target.*

- **Ability to configure the RADIUS shared secret**
- **Ability to define an authorized NAS**
- **Ability to enable, disable, and determine and modify the behavior of all the security functions of the TOE identified in this EP to the administrator**
- **[selection: Ability to configure the IPsec functionality,**
- **No other functions]**

Assurance Activity
Compliance with the other SFRs in section 4.2 and 4.4 of this EP as well as any applicable SFRs in the appendices is sufficient to demonstrate that the TOE provides sufficient means to manage its authentication server functions.

4.4.4.2 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- **Administrator**

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the following conditions

- The Administrator role shall be able to administer the TOE locally;
- The Administrator role shall be able to administer the TOE remotely;

are satisfied.

Assurance Activity	
TSS	The evaluator shall examine the TSS to verify that it describes the administrator role and the powers granted to and limitations of the role.
AGD	The evaluator shall review the operational guidance to ensure that it contains instructions for administering the TOE both locally and remotely, including any configuration that needs to be performed on the client for remote administration.
Test	In the course of performing the testing activities for the evaluation, the evaluator shall use all supported interfaces, although it is not necessary to repeat each test involving an administrative action with each interface. The evaluator shall ensure, however, that each supported method of administering the TOE that conforms to the requirements of this EP be tested; for instance, if the TOE can be administered through a local hardware interface; SSH; and TLS/HTTPS; then all three methods of administration must be exercised during the evaluation team's test activities.

4.4.5 Protection of the TSF (FPT)

4.4.5.1 FPT_TST_EXT.1 TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [selection: during initial start-up (on power on), periodically during normal operation, at the request of the authorized user, at the conditions

[*assignment: conditions under which self-tests should occur*]] to demonstrate the correct operation of the TSF: [*assignment: list of self-tests run by the TSF*].

Assurance Activity	
TSS	The evaluator shall examine the TSS to ensure that it details the self-tests that are run by the TSF on start-up; this description should include an outline of what the tests are actually doing (e.g., rather than saying "memory is tested", a description similar to "memory is tested by writing a value to each memory location and reading it back to ensure it is identical to what was written" shall be used). The evaluator shall ensure that the TSS makes an argument that the tests are sufficient to demonstrate that the TSF is operating correctly.
AGD	The evaluator shall also ensure that the operational guidance describes the possible errors that may result from such tests, and actions the administrator should take in response; these possible errors shall correspond to those described in the TSS.
Test	Future versions of this EP will mandate a clearly defined minimum set of self-tests. For this version of the EP it is expected that at least the following tests are performed: a) Verification of the integrity of the firmware and executable software of the TOE b) Verification of the correct operation of the cryptographic functions necessary to fulfill any of the SFRs. Although formal compliance is not mandated, the self-tests performed should aim for a level of confidence comparable to: a) FIPS 140-2, chap. 4.9.1, Software/firmware integrity test for the verification of the integrity of the firmware and executable software. b) FIPS 140-2, chap. 4.9.1, Cryptographic algorithm test for the verification of the correct operation of cryptographic functions. Alternatively, national requirements of any CCRA member state for the security evaluation of cryptographic functions should be considered as appropriate. The evaluator shall verify that the self-tests described above are either carried out during initial start-up and that the developer has justified any deviation from this (if applicable).

5 Security Assurance Requirements

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

Appendix A: Rationale

In this EP, the focus in the initial sections of the document is to use a narrative presentation in an attempt to increase the overall understandability of the threats addressed by authentication servers; the methods used to mitigate those threats; and the extent of the mitigation achieved by compliant TOEs. This presentation style does not readily lend itself to a formalized evaluation activity, so this section contains the tabular artifacts that can be used for the evaluation activities associated with this document.

A.1 Security Problem Definition

A.1.1 Assumptions

The specific conditions listed below are assumed to exist in the TOE's Operational Environment. These assumptions are in addition to those defined in the base PPs and include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table A-1: TOE Assumptions

Assumption Name	Assumption Definition
A.NAS	It is assumed that the TOE is connected to a Network Access Server (NAS) located in the Operational Environment that transmits authentication requests to it.

A.1.2 Threats

The threats listed below are addressed by authentication servers. Note that these threats are in addition to those defined in the base PPs, all of which apply to authentication servers. In other words, an authentication server that is a software application will face the same threats that a dedicated hardware device will face. The only difference is the extent to which the authentication server relies on an underlying platform rather than its own functionality to mitigate the threats.

Table A-2: Threats

Threat Name	Threat Definition
T.FALSE_ENDPOINTS	A malicious actor may falsely impersonate the TOE or the NAS in order to cause the TOE to operate in an insecure manner or to extract security-relevant data from the TOE or its Operational Environment.
T.INVALID_USERS	A malicious user may supply incorrect credential data or an otherwise invalid authentication request that is approved or ignored by the TSF such that protected resources in the Operational Environment are subject to unauthenticated access.

A.1.3 Organizational Security Policies

There are no organizational security policies defined for this EP.

A.1.4 Security Problem Definition Correspondence

The following table serves to map the threats and assumptions defined in this EP to the security objectives also defined or identified in this EP.

Table A-3: Security Problem Definition Correspondence

Threat or Assumption	Security Objectives	SFR
T.FALSE_ENDPOINTS	OE.TRUSTEDNAS	FCS_EAP-TLS_EXT.1, FIA_PSK_EXT.1, FTP_ITC.1
T.INVALID_USERS	O.RADIUSCOMPLIANT, O.USERAUTH	FCO_NRO.1, FCO_NRR.1, FCS_RADIUS_EXT.1, FCS_RADSEC_EXT.1, FIA_AFL.1 (see 4.3.3.1), FIA_UAU.6, FTA_TSE.1
A.NAS	OE.NAS	N/A

A.2 Security Objectives

A.2.1 Security Objectives for the TOE

The following table contains security objectives specific to authentication servers.

Table A-4: Security Objectives for the TOE

Security Objective Name	Security Objective Definition
O.RADIUSCOMPLIANT	The TOE shall implement the RADIUS protocol in accordance with applicable RFCs.
O.TRUSTEDNAS	The TOE shall provide mechanisms to facilitate mutual authentication with a NAS in the Operational Environment.
O.USERAUTH	The TOE shall provide a mechanism to assess RADIUS authentication requests and respond with accept, reject, or challenge decisions based on the applicable data that is supplied in the request.

A.2.2 Security Objectives for the Operational Environment

The following table contains security objectives specific to the operational environments for authentication servers.

Table A-5: Security Objectives for the Operational Environment

Security Objective Name	Security Objective Definition
OE.NAS	Authentication requests that are provided to the TOE for validation are centrally collected by a NAS and transmitted to the TOE through this component.

A.2.3 Security Objective Correspondence

The correspondence between the Security Functional Requirements (SFRs) and Security Objectives identified or defined in this EP is provided in section 3.

A.3 Rationale for Security Functional Requirements

Table A-6: Rationale for Explicitly Stated Requirements

SFR	Rationale
FCS_EAP-TLS_EXT.1	This SFR was created to define the EAP-TLS method of mutual authentication that is used by the TOE and the NAS to authenticate one another.
FCS_RADIUS_EXT.1	This extended SFR was created to correctly specify the TOE's ability to implement the RADIUS protocol.
FCS_RADSEC_EXT.1	This extended SFR was created to correctly specify the TOE's ability to implement TLS over RADIUS.
FIA_PSK_EXT.1	This SFR was created to define the composition of the RADIUS shared secret that secures the channel between the TOE and the NAS.

Table A-7: SFR Dependency Rationale

SFR	Dependency	Rationale
FCS_EAP-TLS_EXT.1	No dependencies	Not Applicable
FCS_RADIUS_EXT.1	No dependencies	Not Applicable
FCS_RADSEC_EXT.1	No dependencies	Not Applicable
FIA_PSK_EXT.1	No dependencies	Not Applicable

Appendix B: Optional Requirements

This section is reserved for optional requirements that can be included at the discretion of the ST author. If the TOE provides capabilities that are governed by any of these optional requirements, the ST author shall include the relevant requirements in the ST. However, a valid TOE does not need to perform the functions described by these requirements and the ST author can appropriately omit any function that the TOE described by the ST does not provide.

B.1 Identification and Authentication (FIA)

B.1.1 FIA_UAU.6 Re-authenticating

The following optional SFR is to be included when the TSF provides its own administrative authentication mechanism, including TSF-initiated administrative lockout (defined as FIA_UAU_EXT.2, FIA_UIA_EXT.1, and FTA_SSL_EXT.1 in the NDcPP). The SFR is considered to be optional because a TOE that is implemented as application software may rely on an administrative authentication mechanism that is provided by the underlying platform rather than by the TSF.

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 locking (FTA_SSL), [assignment: *other conditions*], no other conditions].

Assurance Activity	
TSS	The evaluator shall examine the TSS to ensure that re-authentication policies are in place, and that the conditions requiring re-authentication and the actions to be taken when these conditions are met are well-defined.
AGD	The evaluator shall examine the operational guidance to determine that it provides guidance to administrators on the conditions that require user re-authentication, and the actions to be taken by the administrator, if any are required.
Test	The evaluator shall perform the following tests for each method by which remote administrators access the TOE: Test 1: The evaluator shall attempt to change their password as directed by the operational guidance. While making this attempt, the evaluator shall verify that re-authentication is required. Test 2: The evaluator shall attempt to log in under each of the remaining conditions specified in the requirement. While making these attempts, the evaluator shall verify that re-authentication is required.

Appendix C: Selection-Based Requirements

The following section includes SFRs that conditionally apply based on the selections chosen in section 4.

C.1 Cryptographic Support (FCS)

C.1.1 EAP-TLS Protocol

The following SFR shall be included in the ST if any ciphersuites using elliptic curves are selected in FCS_EAP-TLS_EXT.1.1 or FCS_RADSEC_EXT.1.3:

FCS_EAP-TLS_EXT.1.7 The TSF shall present the Supported Elliptic Curves Extension in the Client Hello with the following NIST curves: [selection: secp256r1, secp384r1, secp521r1] and no other curves.

Application Note: *This requirement limits the elliptic curves allowed for authentication and key agreement to the NIST curves from FCS_COP.1(3) and FCS_CKM.1(1) and FCS_CKM.2(1). This extension is required for clients supporting Elliptic Curve ciphersuites.*

Assurance Activity
The activities to follow are outlined under the FCS_EAP-TLS_EXT.1 SFR in Section 4.2.2.1.

C.1.2 FCS_RADSEC_EXT.1 – Extended: RadSec

The following SFRs shall be included in the ST if RadSec is selected in FTP_ITC.1

FCS_RADSEC_EXT.1.1 – The TSF shall implement RadSec as specified in RFC 6614, to communicate securely with a NAS.

FCS_RADSEC_EXT.1.2 – The TSF shall perform peer authentication using [selection: X.509v3 certificates, pre-shared keys].

FCS_RADSEC_EXT.1.3 – The TSF shall implement TLS version 1.1 or greater supporting the following cryptosystems:

[selection: Mandatory ciphersuites for X509v3 certificates:

- TLS_RSA_WITH_AES_128_CBC_SHA

,

Mandatory ciphersuites for pre-shared keys:

- TLS_PSK_WITH_AES_128_CBC_SHA]

[selection: Optional ciphersuites for X509v3 certificates:

[selection:

- TLS_RSA_WITH_AES_256_CBC_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA256
- TLS_RSA_WITH_AES_256_CBC_SHA256
- TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA
- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256

- TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384
- TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA
- no other ciphersuites

],
Optional ciphersuites for pre-shared keys:
[selection:

- TLS_PSK_WITH_AES_256_CBC_SHA
- TLS_DHE_PSK_WITH_AES_128_CBC_SHA
- TLS_DHE_PSK_WITH_AES_256_CBC_SHA
- TLS_RSA_PSK_WITH_AES_128_CBC_SHA
- TLS_RSA_PSK_WITH_AES_256_CBC_SHA
- no other ciphersuites

]]

FCS_RADSEC_EXT.1.4 If an optional ciphersuite for pre-shared keys is selected in FCS_RADSEC_EXT.1.3, then the TSF shall be able to [selection: accept, generate using the random bit generator specified in FCS_RBG_EXT.1] bit-based pre-shared keys.

Application Note: *The ciphersuites to be tested in the evaluated configuration are limited by this requirement. The ST author should select the optional ciphersuites that are supported; if there are no ciphersuites supported other than the mandatory suites, then "None" should be selected. It is necessary to limit the ciphersuites that can be used in an evaluated configuration administratively on the server in the test environment. The Suite B algorithms listed above (RFC 6460) are the preferred algorithms for implementation. TLS_RSA_WITH_AES_128_CBC_SHA is required in order to ensure compliance with RFC 5246 and RFC 6614.*

Assurance Activity	
TSS	The evaluator shall verify that the TSS description includes the use of RADIUS over TLS, as described in RFC 6614. If the TOE supports X.509v3 certificates, the evaluator shall ensure that the TSS description includes the use of client-side certificates for TLS mutual authentication. The evaluator shall also verify that the TSS describes how the DN or SAN in the certificate is compared to the expected identifier. The evaluator shall check the description of the implementation of this protocol in the TSS to ensure that the ciphersuites supported are specified. The evaluator shall check the TSS to ensure that the ciphersuites specified are identical to those listed for this component. The evaluator shall also verify that the TSS contains a description of the denial of old SSL and TLS versions. If an optional ciphersuite for pre-shared keys is selected in FCS_RADSEC_EXT.1.3, then the evaluator shall examine the TSS to ensure it describes the process by which the bit-based pre-shared keys are generated (if the TOE supports this functionality), and confirm that this process uses the RBG specified in FCS_RBG_EXT.1.
AGD	The evaluator shall verify that any configuration necessary to meet the requirement must be contained in the guidance. The evaluator shall verify that the guidance includes instructions for configuring certificates for TLS mutual authentication. If the DN is not compared automatically to the Domain Name or IP address, username, or

	email address, then the evaluator shall ensure that the guidance includes configuration of the expected DN or the directory server for the connection. The evaluator shall also check the guidance documentation to ensure that it contains instructions on configuring the TOE so that TLS conforms to the description in the TSS (for instance, the set of ciphersuites advertised by the TOE may have to be restricted to meet the requirements). If an optional ciphersuite for pre-shared keys is selected in FCS_RADSEC_EXT.1.3, then the evaluator shall confirm the operational guidance contains instructions for either entering bit-based pre-shared keys, or generating a bit-based pre-shared key (or both).
Test	The evaluator shall perform the following tests: Test 1: The evaluator shall attempt to establish a non-TLS RADIUS session, and verify that the TOE does not process such requests, and that it signals rejection to the originator. Test 2: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall configure the server to send a certificate request to the client and shall attempt a connection without sending a certificate from the client. The evaluator shall verify that the connection is denied. Test 3: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall configure the server to send a certificate request to the client without the supported_signature_algorithm used by the client's certificate. The evaluator shall attempt a connection using the client certificate and verify that the connection is denied. Test 4: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall demonstrate that using a certificate without a valid certification path results in the function failing. Using the administrative guidance, the evaluator shall then load a certificate or certificates needed to validate the certificate to be used in the function, and demonstrate that the function succeeds. The evaluator then shall delete one of the certificates, and show that the function fails. Test 5: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall present a client certificate that does not chain to one of the Certificate Authorities (either a Root or Intermediate CA) in the server's Certificate Request message. The evaluator shall verify that the attempted connection is denied. Test 6: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall present a client certificate with the Client Authentication purpose in the extendedKeyUsage field and verify that the server accepts the attempted connection. The evaluator shall repeat this test without the Client Authentication purpose and shall verify that the server denies the connection. Ideally, the two certificates should be identical except for the Client Authentication purpose. Test 7: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall perform the following modifications to the traffic: o Modify a byte in the client's certificate. The evaluator shall verify that the server rejects the connection. o Modify a byte in the client's Certificate Verify handshake message. The evaluator shall verify that the server rejects the connection. Test 8: [conditional] If 'X.509v3 certificates' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall send a client certificate with an identifier that does not match an expected identifier and verify that the server denies the connection. Test 9: [conditional] If 'pre-shared keys' is selected in FCS_RADSEC_EXT.1.2, the evaluator shall generate an

	invalid key and demonstrate that a client cannot successfully complete a protocol negotiation using this key. Test 10: The evaluator shall establish a TLS connection using each of the ciphersuites specified by the requirement. This connection may be established as part of the establishment of a higher-level protocol, e.g., as part of an HTTPS session. It is sufficient to observe the successful negotiation of a ciphersuite to satisfy the intent of the test; it is not necessary to examine the characteristics of the encrypted traffic in an attempt to discern the ciphersuite being used (for example, that the cryptographic algorithm is 128-bit AES and not 256-bit AES). Test 11: The evaluator shall send a Client Hello to the server with a list of ciphersuites that does not contain any of the ciphersuites in the server's ST and verify that the server denies the connection. Additionally, the evaluator shall send a Client Hello to the server containing only the TLS_NULL_WITH_NULL_NULL ciphersuite and verify that the server denies the connection. Test 12: The evaluator shall use a client to send a key exchange message in the TLS connection that the does not match the server-selected ciphersuite (for example, send an ECDHE key exchange while using the TLS_RSA_WITH_AES_128_CBC_SHA ciphersuite or send a RSA key exchange while using one of the ECDSA ciphersuites.) The evaluator shall verify that the TOE disconnects after the receiving the key exchange message. Test 13: The evaluator shall perform the following modifications to the traffic: o Modify at a byte in the client's nonce in the Client Hello handshake message, and verify that the server rejects the client's Certificate Verify handshake message (if using mutual authentication) or that the server denies the client's Finished handshake message. o Modify the signature block in the Client's Key Exchange handshake message, and verify that the server rejects the client's Certificate Verify handshake message (if using mutual authentication) or that the server denies the client's Finished handshake message. o Modify a byte in the Client Finished handshake message, and verify that the server rejects the connection and does not send any application data. o After generating a fatal alert by sending a Finished message from the client before the client sends a ChangeCipherSpec message, send a Client Hello with the session identifier from the previous test, and verify that the server denies the connection. o Send a garbled message from the client after the client has issued the ChangeCipherSpec message and verify that the Server denies the connection. Test 14: The evaluator shall send a Client Hello requesting a connection with version SSL 1.0 and verify that the server denies the connection. The evaluator shall repeat this test with SSL 2.0, SSL 3.0, TLS 1.0, and any non-selected TLS versions. Test 15: [conditional] If any optional ciphersuites using pre-shared keys are selected in FCS_RADSEC_EXT.1.3 and the TOE does not generate bit-based pre-shared keys, the evaluator shall obtain a bit-based pre-shared key of the appropriate length and enter it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key. Test 16: [conditional] If any optional ciphersuites using pre-shared keys are selected in FCS_RADSEC_EXT.1.3 and the TOE does generate bit-based pre-shared keys, the evaluator shall generate a bit-based pre-shared key of the appropriate length and use it according to the instructions in the operational guidance. The evaluator shall then demonstrate that a successful protocol negotiation can be performed with the key.

Appendix D: Objective Requirements

This section is reserved for requirements that are not currently prescribed by this EP but are expected to be included in future versions of the EP. Vendors planning on having evaluations performed against future products are encouraged to plan for these objective requirements to be met.

There are no objective requirements currently defined for this EP.