Supporting Document
Mandatory Technical Document

PP-Module for Email Clients
Version: 2.0
2015-06-18
National Information Assurance Partnership

Foreword

This is a Supporting Document (SD), intended to complement the Common Criteria version 3 and the associated Common Evaluation Methodology for Information Technology Security Evaluation.

SDs may be “Guidance Documents”, that highlight specific approaches and application of the standard to areas where no mutual recognition of its application is required, and as such, are not of normative nature, or “Mandatory Technical Documents”, whose application is mandatory for evaluations whose scope is covered by that of the SD. The usage of the latter class is not only mandatory, but certificates issued as a result of their application are recognized under the CCRA.

Technical Editor:
National Information Assurance Partnership (NIAP)

Document history:

Version	Date	Comment
v 1.0	2014-04-01	Release - Email Client Protection Profile
v 2.0	2015-06-18	Update as Extended Package of the Protection Profile for Application Software
v 2.0	2015-06-18	Application Software Module for Email Clients

General Purpose:
The purpose of this SD is to define evaluation methods for the functional behavior of Email Clients products.

Acknowledgements:
This SD was developed with support from NIAP Email Clients Technical Community members, with representatives from industry, government agencies, Common Criteria Test Laboratories, and members of academia.

Table of Contents

1 Introduction

1.1 Technology Area and Scope of Supporting Document

The scope of the PP-Module for Email Clients is to describe the security functionality of Email Clients products in terms of [CC] and to define functional and assurance requirements for them. The PP-Module is intended for use with the following Base-PP:

- Protection Profile for Application Softwares, Version

This SD is mandatory for evaluations of TOEs that claim conformance to a PP-Configuration that includes the PP-Module for :

- Email Clients, Version 2.0

As such it defines Evaluation Activities for the functionality described in the PP-Module as well as any impacts to the Evaluation Activites to the Base-PP(s) it modifies.

Although Evaluation Activities are defined mainly for the evaluators to follow, in general they also help developers to prepare for evaluation by identifying specific requirements for their TOE. The specific requirements in Evaluation Activities may in some cases clarify the meaning of Security Functional Requirements (SFR), and may identify particular requirements for the content of Security Targets (ST) (especially the TOE Summary Specification), user guidance documentation, and possibly supplementary information (e.g. for entropy analysis or cryptographic key management architecture).

1.2 Structure of the Document

Evaluation Activities can be defined for both SFRs and Security Assurance Requirements (SAR), which are themselves defined in separate sections of the SD.

If any Evaluation Activity cannot be successfully completed in an evaluation, then the overall verdict for the evaluation is a 'fail'. In rare cases there may be acceptable reasons why an Evaluation Activity may be modified or deemed not applicable for a particular TOE, but this must be approved by the Certification Body for the evaluation.

In general, if all Evaluation Activities (for both SFRs and SARs) are successfully completed in an evaluation then it would be expected that the overall verdict for the evaluation is a ‘pass’. To reach a ‘fail’ verdict when the Evaluation Activities have been successfully completed would require a specific justification from the evaluator as to why the Evaluation Activities were not sufficient for that TOE.

Similarly, at the more granular level of assurance components, if the Evaluation Activities for an assurance component and all of its related SFR Evaluation Activities are successfully completed in an evaluation then it would be expected that the verdict for the assurance component is a ‘pass’. To reach a ‘fail’ verdict for the assurance component when these Evaluation Activities have been successfully completed would require a specific justification from the evaluator as to why the Evaluation Activities were not sufficient for that TOE.

2 Evaluation Activities for SFRs

The EAs presented in this section capture the actions the evaluator performs to address technology specific aspects covering specific SARs (e.g. ASE_TSS.1, ADV_FSP.1, AGD_OPE.1, and ATE_IND.1) – this is in addition to the CEM work units that are performed in Section 3 Evaluation Activities for SARs.

Regarding design descriptions (designated by the subsections labelled TSS, as well as any required supplementary material that may be treated as proprietary), the evaluator must ensure there is specific information that satisfies the EA. For findings regarding the TSS section, the evaluator’s verdicts will be associated with the CEM work unit ASE_TSS.1-1. Evaluator verdicts associated with the supplementary evidence will also be associated with ASE_TSS.1-1, since the requirement to provide such evidence is specified in ASE in the PP.

For ensuring the guidance documentation provides sufficient information for the administrators/users as it pertains to SFRs, the evaluator’s verdicts will be associated with CEM work units ADV_FSP.1-7, AGD_OPE.1-4, and AGD_OPE.1-5.

Finally, the subsection labelled Tests is where the authors have determined that testing of the product in the context of the associated SFR is necessary. While the evaluator is expected to develop tests, there may be instances where it is more practical for the developer to construct tests, or where the developer may have existing tests. Therefore, it is acceptable for the evaluator to witness developer-generated tests in lieu of executing the tests. In this case, the evaluator must ensure the developer’s tests are executing both in the manner declared by the developer and as mandated by the EA. The CEM work units that are associated with the EAs specified in this section are: ATE_IND.1-3, ATE_IND.1-4, ATE_IND.1-5, ATE_IND.1-6, and ATE_IND.1-7.

2.1 Protection Profile for Application Softwares

The EAs defined in this section are only applicable in cases where the TOE claims conformance to a PP-Configuration that includes the ApSo PP.

2.1.1 Modified SFRs

The PP-Module does not modify any requirements when the ApSo PP is the base.

2.1.2 Additional SFRs

Secure/Multipurpose Internet Mail Extensions (S/MIME)

The evaluator verifies that the version of S/MIME implemented by the email client is present in the TSS. The evaluator also verifies that the algorithms supported are specified, and that the algorithms specified are those listed for this component.
The evaluator verifies that the TSS describes the ContentEncryptionAlgorithmIdentifier and whether the required behavior is performed by default or may be configured.
The evaluator verifies that the TSS describes the digestAlgorithm and whether the required behavior is performed by default or may be configured.
The evaluator verifies that the TSS describes the AlgorithmIdentifier and whether the required behavior is performed by default or may be configured.
The evaluator verifies that the TSS describes the retrieval mechanisms for both certificates and certificate revocation as well as the frequency at which these mechanisms are implemented. The evaluator also reviews the Operational Guidance to ensure that it contains instructions on configuring the email client such that it complies with the description in the TSS.
If the TSS indicates that the algorithms in FCS_SMIME_EXT.1.2 must be configured to meet the requirement, the evaluator verifies that the AGD guidance includes the configuration of this ID.
If the TSS indicates that the algorithms in FCS_SMIME_EXT.1.3 must be configured to meet the requirement, the evaluator verifies that the AGD guidance includes the configuration.
If the TSS indicates that the algorithms in FCS_SMIME_EXT.1.4 must be configured to meet the requirement, the evaluator verifies that the AGD guidance includes the configuration of this ID.
If the TSS indicates that the mechanisms in FCS_SMIME_EXT.1.7 are configurable, the evaluator verifies that the AGD guidance includes the configuration of these mechanisms. The evaluator performs the following tests:
These tests can be performed in conjunction with the tests specified in FIA_X509_EXT.1 (defined in) for certificate/certificate chain verification and FDP_NOT_EXT.1.
Test 1: The evaluator both sends and receives a message with no protection (no signature or encryption) and verify that the message is transmitted properly and can be viewed at the receiving agent. This transmission can be performed as part of a number of mechanisms; it is sufficient to observe that the message arrives at the intended recipient with the same content as when sent.
Test 2: The evaluator both sends and receives a signed message using each of the algorithms specified in the ST corresponding to the requirement and verify that the signature is valid for both received and sent messages. After verifying the signatures are valid, the evaluator sends a signed message using each of the algorithms specified in the ST and use a maninthemiddle tool to modify at least one byte of the message such that the signature is no longer valid. This can be done by modifying the content of the message over which the signature is calculated or by modifying the signature itself. The evaluator verifies that the received message fails the signature validation check.
Test 3: The evaluator both sends and receives an encrypted message using each of the algorithms specified in the ST. The evaluator verifies that the contents are encrypted in transit and that the received message decrypts. After verifying the message decrypts, the evaluator sends an encrypted message using each of the algorithms specified in the ST and use a maninthemiddle tool to modify at least one byte of the message such that the encryption is no longer valid. The evaluator verifies that the received message fails to decrypt.
Test 4: The evaluator both sends and receives a message that is both signed and encrypted. In addition, the evaluator uses a man-in-the-middle tool to modify at least one byte of the message such that the encryption and signature are no longer valid. The evaluator verifies that the received message fails to decrypt, fails the signature validation check, and/or both.
Test 5: The evaluator sends a signed message to the email client using a signature algorithm not supported according to the digestAlgorithm ID (e.g., SHA1). The evaluator verifies that the email client provides a notification that the contents cannot be verified because the signature algorithm is not supported.
Test 6: The evaluator sends an encrypted message to the email client using an encryption algorithm not supported according to the AlgorithmIdentifier field. The evaluator verifies that the email client does not display/decrypt the contents of the message.
Test 7: The evaluator sends the email client a message signed by a certificate without the digitalSignature bit set. The evaluator verifies that the email client notifies the user that the signature is invalid.
Test 8: The evaluator sends the email client a message signed by a certificate without the Email Protection purpose in the extendedKeyUsage. The evaluator verifies that the email client notifies the user that the signature is invalid.
Test 9: The evaluator verifies that the email client uses OCSP or downloads the CRL at the assigned

frequency.
TSS
Guidance
Tests

Protection of Key and Key Material

The evaluator verifies the TSS for a high level description of method used to protect keys stored in nonvolatile memory.
The evaluator verifies the TSS to ensure it describes the storage location of all keys and the protection of all keys stored in nonvolatile memory. The description of the key chain shall be reviewed to ensure FCS_COP_EXT.2 is followed for the storage of wrapped or encrypted keys in nonvolatile memory and plaintext keys in nonvolatile memory meet one of the criteria for storage.
TSS

Cryptographic Key Destruction

If the platform provides the key destruction, then the evaluator examines the TSS to verify that it describes how the key destruction functionality is invoked.
If the application invokes key destruction, the evaluator checks to ensure the TSS describes each of the secret keys (keys used for symmetric encryption and/or data authentication), private keys, and CSPs used to generate key; when they are zeroized (for example, immediately after use, on system shutdown, etc.); and the type of zeroization procedure that is performed (overwrite with zeros, overwrite three times with random pattern, etc.). If different types of memory are used to store the materials to be protected, the evaluator checks to ensure that the TSS describes the zeroization procedure in terms of the memory in which the data are stored (for example, "secret keys stored on a drive are zeroized by overwriting once with zeros, while secret keys stored on the internal hard drive are zeroized by overwriting three times with a random pattern that is changed before each write"). The following test is only for key destruction provided by the email client:
Test 1: For each type of authorization service, encryption mode and encryption operation, a known authorization factor, and chain of keys must be provided to the evaluator with an associated ciphertext data set (e.g. if a passphrase is used to create a intermediate key, then the ciphertext containing the encrypted key as well as the intermediate key itself must be provided to the evaluator.) The evaluator will use the email client in conjunction with a debugging or forensics utility to attempt to authorize themselves, resulting in the generation of a key or decryption of a key. The evaluator will ascertain from the TSS what the vendor defines as "no longer needed" and execute the sequence of actions via the email client to invoke this state. At this point, the evaluator should take a dump of volatile memory and search the retrieved dump for the provided authorization credentials or keys (e.g. if the password was "PaSSw0rd", perform a string search of the forensics dump for "PaSSw0rd"). The evaluator must document each command, program or action taken during this process, and must confirm that no plaintext keying material resides in volatile memory. The evaluator must perform this test three times to ensure repeatability. If during the course of this testing the evaluator finds that keying material remains in volatile memory, they should be able to identify the cause (i.e. execution of the grep command for "PaSSw0rd" caused a false positive) and document the reason for failure to comply with this requirement. The evaluator will repeat this same test, but looking for keying material in nonvolatile memory.
TSS
Tests

Key Chaining

The evaluator verifies the TSS* describes a high level description of the key hierarchy for all authorizations methods that are used to protect the encryption keys. The evaluator will examine the TSS to ensure it describes the key chain in detail. The description of the key chain shall be reviewed to ensure it maintains a chain of keys using key wrap that meets FCS_COP_EXT.2. The evaluator verifies the TSS* to ensure that it describes how the key chain process functions, such that it does not expose any material that might compromise any key in the chain. A high-level description should include a diagram illustrating the key hierarchy implemented and detail where all keys and keying material is stored or what it is derived from. The evaluator will examine the key hierarchy to ensure that at no point the chain could be broken without a cryptographic exhaust or knowledge of the key within the chain and the effective strength of the data encryption key is maintained throughout the key chain.
*If necessary, this information could be contained in a proprietary document and not appear in the TSS.
TSS

Notification of S/MIME Status

The evaluator will ensure that the TSS describes notifications of S/MIME status, including whether S/MIME status is also indicated upon viewing a list of emails. The evaluator verifies that the AGD guidance provides a description (with appropriate visual figures) of the S/MIME status notification(s), including how each of the following are indicated: encryption, verified and validated signature, and unverified and unvalidated signature. The evaluator will perform the following tests and may perform them in conjunction with the tests for FCS_SMIME_EXT.1:

- Test 1: The evaluator will send the client an unencrypted and unsigned email and verify that no notifications are present upon viewing.

- Test 2: The evaluator will send the client an encrypted email and verify that the encrypted notification is present upon viewing.
- Test 3: The evaluator will send the client a valid signed email and verify that the signed notification is present upon viewing.
- Test 4: The evaluator will send the client an invalid signed email (for example, using a certificate that does not contain the correct email address or a certificate that does not chain to the root store) and verify that the invalid signature notification is present upon viewing.

TSS
Guidance
Tests

S/MIME

The evaluator verifies that the TSS contains a description of the S/MIME implementation and its use to protect mail from undetected modification using digital signatures and unauthorized disclosure using encryption. The evaluator verifies that the TSS describes whether signature verification and decryption occur at receipt or viewing of the message contents, and whether messages are stored with their S/MIME envelopes. The evaluator will ensure that the AGD guidance includes instructions for configuring a certificate for S/MIME use and instructions for signing and encrypting email. Tests for this element are performed in conjunction with tests for FCS_SMIME_EXT.1 and FDP_NOT_EXT.1
TSS
Guidance
Tests

X509 Authentication and Encryption

The evaluator checks the TSS to ensure that it describes how the email client chooses which certificates to use so that the email client can use the certificates.
The evaluator will examine the TSS to confirm that it describes the behavior of the email client when a connection cannot be established during the validity check of a certificate used in establishing a trusted channel and protecting email. The evaluator verifies that the administrative guidance contains any necessary instructions for configuring the operating environment so that the email client can use the certificates. The evaluator will perform the following tests:

- Test 1: The evaluator will perform Test 1 for each function listed in FIA_X509_EXT.2.1 in that requires the use of certificates. The evaluator will demonstrate that using a certificate without a valid certification path results in the function failing. The evaluator will then load into the platform's root store any certificates needed to validate the certificate to be used in the function, and demonstrate that the function succeeds.
- Test 2: The evaluator will demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a nonTOE IT entity. The evaluator will then manipulate the environment so that the email client is unable to verify the validity of the certificate, and observe that the action selected in FIA_X509_EXT.2.2 in is performed. If the selected action is administratorconfigurable, then the evaluator will follow the operational guidance to determine that all supported administrator configurable options behave in their documented manner.

TSS
Guidance
Tests

Management of Functions Behavior

The evaluation activities for this component will be driven by the selections made by the ST author. If a capability is not selected in the ST, the noted evaluation activity does not need to be performed. The evaluator verifies that the TSS describes those management functions which may only be configured by the email client platform administrator and cannot be overridden by the user when set according to policy.
Change Password: The evaluator will examine the Operational Guidance to ensure that it describes how the password/passphrase-based authorization factor is to be changed.
Disable Key Recovery: If the email client supports key recovery, this must be stated in the TSS. The TSS shall also describe how to disable this functionality. This includes a description of how the recovery material is provided to the recovery holder.
Cryptographic Configuration: The evaluator will determine from the TSS for other requirements (FCS_*) what portions of the cryptographic functionality are configurable. The evaluator will examine the operational guidance to verify that it includes instructions for an email client platform administrator to configure the functions listed in FMT_MOF_EXT.1.1.
Disable Key Recovery: If the email client supports key recovery, the guidance for disabling this capability shall be described in the AGD documentation.
Cryptographic Configuration: The evaluator will review the AGD documentation to determine that there are instructions for manipulating all of the claimed mechanisms. The evaluator will perform the following tests:

- Test 1: The evaluator verifies that functions perform as intended by enabling, disabling, and configuring the functions.
- Test 2: The evaluator will set management functions which are controlled by the (enterprise)

administrator and cannot be overridden by the user. The evaluator will apply these functions to the client, attempt to override each setting as the user, and ensure that the email client does not permit it.
- Test 3: Disable Key Recovery: If the email client provides key recovery capability, then the evaluator will devise a test that ensures that the key recovery capability has been or can be disabled following the guidance provided by the vendor

TSS
Guidance
Tests

Support for Only Trusted Add-ons

The evaluator verifies that the TSS describes whether the email client is capable of loading trusted add-ons. The evaluator will examine the operational guidance to verify that it includes instructions on loading trusted add-on sources. The evaluator will perform the following test:

- Test 1: The evaluator will create or obtain an untrusted add-on and attempt to load it. The evaluator verifies that the untrusted add-on is rejected and cannot be loaded.

TSS
Guidance
Tests

Inter-TSF Trusted Channel

The evaluator will examine the TSS to determine that it describes the details of the email client connecting to a Mail Transfer Agent in terms of the trusted connection (i.e., TLS) according to FTP_DIT_EXT.1 in , along with email client-specific options or procedures that might not be reflected in the specification. The evaluator will confirm that the operational guidance contains instructions for establishing the connection to the Mail Transfer Agent. The evaluator will also perform the following tests:

- Test 1: The evaluators shall ensure that the email client is able to initiate communications using any selected or assigned protocols specified in the requirement over TLS, setting up the connections as described in the operational guidance and ensuring that communication is successful.
- Test 2: The evaluators shall ensure that the email client is able to initiate communications with a Mail Transfer Agent using SMTP and any assigned protocols specified in the requirement over TLS, setting up the connections as described in the operational guidance and ensuring that communication is successful.
- Test 3: The evaluator will ensure, for each communication channel with an authorized IT entity in tests 1 and 2, the channel data is not sent in plaintext. To perform this test, the evaluator will use a sniffer and a packet analyzer. The packet analyzer must indicate that the protocol in use is TLS.

TSS
Guidance
Tests

2.2 TOE SFR Evaluation Activities

The PP-Module does not define any mandatory requirements (i.e. Requirements that are included in every configuration regardless of the PP-Bases selected).

2.3 Evaluation Activities for Optional SFRs

A password/passphrase used to generate a password authorization factor shall enable up to positive integer of 64 or more characters in the set of upper case characters lower case characters numbers special characters: !, @, #, $, %, ^, &, *, (,) other supported special characters and shall perform [Password-based Key Derivation Functions] in accordance with a specified cryptographic algorithm HMAC-SHA-256 HMAC-SHA-384 HMAC-SHA-512 with positive integer of 4096 or more iterations, and output cryptographic key sizes of 128 bits 256 bits that meet NIST SP 800-132. The password/passphrase is represented on the host machine as a sequence of characters whose encoding depends on the email client and the underlying OS. This sequence must be conditioned into a string of bits that is to be used as a key of equivalent size to the rest of the key chain. This password/passphrase must be conditioned into a string of bits that forms the submask to be used as input into a key. Conditioning can be performed using one of the identified hash functions or the process described in NIST SP 800-132; the method used is selected by the Author. SP 800-132 requires the use of a pseudo-random function (PRF) consisting of HMAC with an approved hash function. The author selects the hash function used, also includes the appropriate requirements for HMAC and the hash function. Appendix A of SP 800-132 recommends setting the iteration count in order to increase the computation needed to derive a key from a password and, therefore, increase the workload of performing a password recovery attack. However, for this , a minimum iteration count of 4096 is required in order to ensure that twelve bits of security is added to the password/passphrase value. A significantly higher value is recommended to ensure optimal security. There are two aspects of this component that require evaluation: passwords/passphrases of the length specified in the requirement (at least 64 characters) are supported, and that the characters that are input are subject to the selected conditioning function. These activities are separately addressed in the tests below. The evaluator verifies that the TSS section specifies the capability that exists to accept passwords/passphrases with the minimum number of characters specified in the ST in this assignment statement. The evaluator examines the password hierarchy TSS to ensure that the formation of all keys is described and that the key sizes match that described by the ST author. The evaluator checks that the TSS describes the method by which the password/passphrase is first encoded and then fed to the SHA algorithm. The settings for the algorithm (padding, blocking, etc.) shall be described, and the evaluator verifies that these are supported by the selections in this component as well as the selections concerning the hash function itself. The evaluator verifies that the TSS contains a description of how the output of the hash function is used to form the submask that will be input into the function. For the NIST SP 800132based conditioning of the password/passphrase, the required evaluation activities will be performed when doing the evaluation activities for the appropriate requirements (FCS_COP.1.1(4) in). If any manipulation of the key is performed in forming the submask that will be used to form the key, that process shall be described in the TSS. No explicit testing of the formation of the submask from the input password is required. The evaluators shall check the Operational Guidance to determine that there are instructions for guidance on how to generate large passwords/passphrases external to the email client and instructions for how to configure the password/passphrase length and optional complexity settings (note to Management section). This is important because many default settings for passwords/passphrases will not meet the necessary entropy needed as specified in this EP. The evaluator will also perform the following tests: Test 1: Ensure that the email client supports passwords/passphrases of 64 characters Test 2: Try entering a password/passphrase less than 64 characters. Test 3: If the email client supports a password/passphrase length up to a maximum number of characters, n (which would be greater than 64), then ensure that the email client will not accept more than n characters. Conditioning: No explicit testing of the formation of the authorization factor from the input password/passphrase is required. The evaluator verifies that the iteration count for PBKDFs performed by the email client comply with NIST SP 800132 by ensuring that the TSS contains a description of the estimated time required to derive key material from passwords and how the email client increases the computation time for passwordbased key derivation (including but not limited to increasing the iteration count). This SFR defines how clients generate salts for cryptographic operations. It does not impact functionality described by the PP. The email client shall only use salts that are generated by a RNG as specified in FCS_RBG_EXT.1 RNG provided by the host platform The salt must be random. The evaluator will ensure the TSS describes how salts are generated. The evaluator will confirm that the salt is generated using an described in FCS_RBG_EXT.1 in or by the Operational Environment. If external function is used for this purpose, the TSS should include the specific API that is called with inputs. If the email client is relying on random bit generation from the host platform, the evaluator verifies the TSS includes the name/manufacturer of the external RBG and describes the function call and parameters used when calling the external DRBG function. If different external RBGs are used for different platforms, the TSS identifies each RBG for each platform. Also, the TSS includes a short description of the vendor's assumption for the amount of entropy seeding the external DRBG. This SFR defines how clients generate nonces for cryptographic operations. It does not impact functionality described by the PP. The email client shall only use unique nonces with a minimum size of [64] bits. Nonces must be unique. The evaluator will ensure the TSS describes how nonces are created uniquely. This SFR defines how clients generate IVs for cryptographic operations. It does not impact functionality described by the PP. The email client shall create IVs in the following manner: CBC: IVs shall be non-repeating CCM: IV shall be non-repeating XTS: No IV. Tweak values shall be non-negative integers, assigned consecutively, and starting at an arbitrary non-negative integer GCM: IV shall be non-repeating. The number of invocations of GCM shall not exceed 2^32 for a given secret key. FCS_IVG_EXT.1.1 specifies how the IV should be handled for each encryption mode. CBC, XTS, and GCM are allowed for AES encryption of the data. AES-CCM is an allowed mode for Key Wrapping. The evaluator will ensure the TSS describes how IVs and tweaks are handled (based on the AES mode). The evaluator will confirm that the IVs and tweaks meet the stated requirements. If the platform provides the IV generation, then the evaluator will examine the TSS to verify that it describes how the IV generation is invoked. This SFR defines how clients display URIs in embedded links It does not impact functionality described by the PP. The email client shall display the full Uniform Resource Identifier (URI) of any embedded links. Embedded links are HTML URI objects which may have a tag (such as a word, phrase, icon, or picture) that obfuscates the URI of the link. The intent of this requirement is to de-obfuscate the link. The URI may be displayed as a "mouse-over" event or may be rendered next to the tag. The evaluator verifies that the TSS includes a description of how embedded links are rendered and the method by which the URI of the link is displayed. The evaluator will ensure that the AGD guidance includes instructions (with any appropriate visual figures) for viewing the URI of an embedded link. The evaluator will perform the following test: Test 1: The evaluator will send the client an HTML message with an embedded link whose tag is not the URI itself (for example, "click here"). The evaluator will view the message and, following the instructions in the AGD guidance, verify that the full URI of the embedded link is displayed. This SFR defines how clients display URIs in embedded links It does not impact functionality described by the PP. The email client shall be capable of operating without storing persistent information to the client platform with the following exceptions: credential information administrator provided configuration information certificate revocation information no exceptions . Any data that persists after the email client closes, including temporary files, is considered to be persistent data. Satisfying this requirement would require the use of a protocol such as IMAP or MAPI. It is not compatible with POP. The evaluator will examine the TSS to determine that it describes all persistent information stored on the platform, and the locations on the platform where these data are stored. The evaluator will confirm that the persistent data described is limited to the data identified in the selection. The evaluator will perform the following tests: Test 1: The evaluator will operate the email client so that several messages, signed, encrypted, and unsigned, are processed. The evaluator will also exercise functionality such as moving messages to folders, writing unsent drafts of messages, etc., as provided by the client. The evaluator will then examine the client platform to determine that the only persistent information stored is that identified in the TSS. This SFR defines functionality to display message content. It does not impact functionality described by the PP. The email client shall have a plaintext-only mode which disables the rendering and execution of HTML JavaScript other embedded content types no embedded content types . Plaintext only mode prevents the automatic downloading, rendering and execution of images, external resources and embedded objects such as HTML or JavaScript objects. addresses configuration of this mode. The ST author must identify all content types supported by the email client through selections and/or assignments. If the email client only supports plaintext only mode, no embedded content types should be selected. The evaluator will ensure that the TSS describes plaintext only mode for sending and receiving messages. The evaluator verifies that the TSS describes whether the email client is capable of rendering and executing HTML or JavaScript. If the email client can render or execute HTML or JavaScript, this description must indicate how the email client handles received messages that contain HTML or JavaScript while in plaintext only mode, and the evaluator will ensure that the description indicates that embedded objects of these types are not rendered or executed and images/external resources are not automatically downloaded. The evaluator will examine the AGD guidance and verify that it contains instructions for enabling plaintext only mode. The evaluator will perform the following tests: Test 1: If HTML is selected in FDP_REN_EXT.1.1, the evaluator will send a message to the client containing HTML embedded objects and shall verify that the HTML renders. The evaluator will then enable plaintext only mode and verify that the HTML does not render. Test 2: If JavaScript is selected in FDP_REN_EXT.1.1, the evaluator will send a message to the client containing JavaScript embedded objects and shall verify that the JavaScript renders and executes. The evaluator will then enable plaintext only mode and verify that the JavaScript does not render or execute.

Cryptographic Key Derivation (Password/Passphrase Conditioning)

The evaluator verifies that the TSS section specifies the capability that exists to accept passwords/passphrases with the minimum number of characters specified in the ST in this assignment statement.
The evaluator examines the password hierarchy TSS to ensure that the formation of all keys is described and that the key sizes match that described by the ST author. The evaluator checks that the TSS describes the method by which the password/passphrase is first encoded and then fed to the SHA algorithm. The settings for the algorithm (padding, blocking, etc.) shall be described, and the evaluator verifies that these are supported by the selections in this component as well as the selections concerning the hash function itself. The evaluator verifies that the TSS contains a description of how the output of the hash function is used to form the submask that will be input into the function. For the NIST SP 800132based conditioning of the password/passphrase, the required evaluation activities will be performed when doing the evaluation activities for the appropriate requirements (FCS_COP.1.1(4) in). If any manipulation of the key is performed in forming the submask that will be used to form the key, that process shall be described in the TSS. No explicit testing of the formation of the submask from the input password is required. The evaluators shall check the Operational Guidance to determine that there are instructions for guidance on how to generate large passwords/passphrases external to the email client and instructions for how to configure the password/passphrase length and optional complexity settings (note to Management section). This is important because many default settings for passwords/passphrases will not meet the necessary entropy needed as specified in this EP. The evaluator will also perform the following tests:

- Test 1: Ensure that the email client supports passwords/passphrases of 64 characters
- Test 2: Try entering a password/passphrase less than 64 characters.
- Test 3: If the email client supports a password/passphrase length up to a maximum number of characters, n (which would be greater than 64), then ensure that the email client will not accept more than n characters. Conditioning: No explicit testing of the formation of the authorization factor from the input password/passphrase is required.

The evaluator verifies that the iteration count for PBKDFs performed by the email client comply with NIST SP 800132 by ensuring that the TSS contains a description of the estimated time required to derive key material from passwords and how the email client increases the computation time for passwordbased key derivation (including but not limited to increasing the iteration count).
TSS
Guidance
Tests

Cryptographic Salt Generation

The evaluator will ensure the TSS describes how salts are generated. The evaluator will confirm that the salt is generated using an described in FCS_RBG_EXT.1 in or by the Operational Environment. If external function is used for this purpose, the TSS should include the specific API that is called with inputs.
If the email client is relying on random bit generation from the host platform, the evaluator verifies the TSS includes the name/manufacturer of the external RBG and describes the function call and parameters used when calling the external DRBG function. If different external RBGs are used for different platforms, the TSS identifies each RBG for each platform. Also, the TSS includes a short description of the vendor's assumption for the amount of entropy seeding the external DRBG.
TSS

Cryptographic Nonce Generation

The evaluator will ensure the TSS describes how nonces are created uniquely.
TSS

Initialization Vector Generation

The evaluator will ensure the TSS describes how IVs and tweaks are handled (based on the AES mode). The evaluator will confirm that the IVs and tweaks meet the stated requirements.
If the platform provides the IV generation, then the evaluator will examine the TSS to verify that it describes how the IV generation is invoked.
TSS

Notification of URI

The evaluator verifies that the TSS includes a description of how embedded links are rendered and the method by which the URI of the link is displayed. The evaluator will ensure that the AGD guidance includes instructions (with any appropriate visual figures) for viewing the URI of an embedded link. The evaluator will perform the following test:

- Test 1: The evaluator will send the client an HTML message with an embedded link whose tag is not the URI itself (for example, "click here"). The evaluator will view the message and, following the instructions in the AGD guidance, verify that the full URI of the embedded link is displayed.

TSS
Guidance
Tests

Storage of Persistent Information

The evaluator will examine the TSS to determine that it describes all persistent information stored on the platform, and the locations on the platform where these data are stored. The evaluator will confirm that the persistent data described is limited to the data identified in the selection. The evaluator will perform the following tests:

- Test 1: The evaluator will operate the email client so that several messages, signed, encrypted, and unsigned, are processed. The evaluator will also exercise functionality such as moving messages to folders, writing unsent drafts of messages, etc., as provided by the client. The evaluator will then examine the client platform to determine that the only persistent information stored is that identified in the TSS.

TSS
Tests

Rendering of Message Content

The evaluator will ensure that the TSS describes plaintext only mode for sending and receiving messages. The evaluator verifies that the TSS describes whether the email client is capable of rendering and executing HTML or JavaScript. If the email client can render or execute HTML or JavaScript, this description must indicate how the email client handles received messages that contain HTML or JavaScript while in plaintext only mode, and the evaluator will ensure that the description indicates that embedded objects of these types are not rendered or executed and images/external resources are not automatically downloaded. The evaluator will examine the AGD guidance and verify that it contains instructions for enabling plaintext only mode. The evaluator will perform the following tests:

- Test 1: If HTML is selected in FDP_REN_EXT.1.1, the evaluator will send a message to the client containing HTML embedded objects and shall verify that the HTML renders. The evaluator will then enable plaintext only mode and verify that the HTML does not render.
- Test 2: If JavaScript is selected in FDP_REN_EXT.1.1, the evaluator will send a message to the client containing JavaScript embedded objects and shall verify that the JavaScript renders and executes. The evaluator will then enable plaintext only mode and verify that the JavaScript does not render or execute.

TSS
Guidance
Tests

2.4 Evaluation Activities for Selection-Based SFRs

This SFR defines how email clients to verify Add-Ons. It does not impact functionality described by the PP. The email client shall provide the ability leverage the platform to provide a means to cryptographically verify add-ons using a digital signature mechanism and published hash no other functions prior to installation and update. The email client shall provide the abilityleverage the platform to query the current version of the add-on. The email client shall prevent the automatic installation of add-ons. The evaluator will examine the TSS to verify that it states that the email client will reject add-ons from untrusted sources. The evaluator will examine the operational guidance to verify that it includes instructions on how to configure the email client with trusted add-on sources. The evaluator will perform the following tests: Test 1: The evaluator will create or obtain an add-on signed by a trusted source and attempt to install it. The evaluator verifies that the signature on the addon is valid and that the add-on can be installed. Test 2: The evaluator will create or obtain an add-on signed with an invalid certificate and attempt to install it. The evaluator verifies that the signed addon is rejected and cannot be installed. Test 3: The evaluator will create or obtain an add-on signed by a trusted source, modify the addon without resigning it, and attempt to install it. The evaluator verifies that the signed add-on is rejected and cannot be installed. This SFR defines an alternate method of transmitting messagess. It does not impact functionality described by the PP. The email client shall implement support for Simple Authentication and Security Layer (SASL) that complies with RFC 4422. SASL is needed if the email implements SMTP to send messages. Clients that do not use SMTP (e.g., ActiveSync or MAPI) would not need to implement support for SASL. The email client shall support the POP3 CAPA and AUTH extensions for the SASL mechanism. The email client shall support the IMAP CAPABILITY and AUTHENTICATE extensions for the SASL mechanism. The email client shall support the SMTP AUTH extension for the SASL mechanism. In order for an email client to support PKI X.509 Certificates for POP3, IMAP and SMTP as required in this document, the client must support the Simple Authentication and Security Layer (SASL) authentication method as described in RFC 4422, the AUTH and CAPA extensions for POP3, as described in RFC 5034, the AUTHENTICATION and CAPABILITY extensions for IMAP, as described in RFC 4959 and the AUTH extension for SMTP, as described in RFC 4954. The evaluator will examine the TSS to determine that it describes the details of the email client connecting to a Mail Transfer Agent in terms of the SASL connection, along with email clientspecific options or procedures that might not be reflected in the specification. The evaluator will confirm that the operational guidance contains instructions for establishing the connection to the Mail Transfer Agent. The evaluator will also perform the following tests: Test 1: The evaluators shall ensure that the email client is able to initiate communications using POP, IMAP and SMTP and requiring SASL, setting up the connections as described in the operational guidance and ensuring that communication is successful. Test 2: The evaluator will ensure, for each communication channel with an authorized IT entity in tests 1, that a valid SASL handshake is performed. To perform this test, the evaluator will use a sniffer and a packet analyzer. The packet analyzer must indicate that the protocol in use is SASL. This SFR defines how clients combine keys. It does not impact functionality described by the PP. The email client shall combine submasks using the following method exclusive OR (XOR) SHA-256 SHA-512 to generate another key. This requirement specifies the way that a product may combine the various submasks by using either an XOR or an approved SHA-hash. If keys are XORed together to form an intermediate key, the TSS section shall identify how this is performed (e.g., if there are ordering requirements, checks performed, etc.). The evaluator will also confirm that the TSS describes how the length of the output produced is at least the same as that of the data encryption key. This SFR defines how clients wrap keys. It does not impact functionality described by the PP. The email client shall use platform-provided functionality to perform Key Wrapping implement functionality to perform Key Wrapping in accordance with a specified cryptographic algorithm AES Key Wrap AES Key Wrap with Padding RSA using the KTS-OAEP-basic scheme RSA using the KTS-OAEP-receiver-confirmation scheme ECC CDH and the cryptographic key size 128 bits (AES) 256 bits (AES) 2048 (RSA) 4096 (RSA) 256-bit prime modulus (ECC CDH) 384-bit prime modulus (ECC CDH) that meet the following: "NIST SP 800-38F" for Key Wrap (section 6.2) and Key Wrap with Padding (section 6.3) "NIST SP 800-56B" for RSA using the KTS-OAEP-basic (section 9.2.3) and KTS-OAEP-receiver-confirmation (section 9.2.4) scheme, "NIST SP 800-56A rev 2" for ECC CDH (sections 5.6.1.2 and 6.2.2.2) . In the first selection, the ST author chooses the entity that performs the decryption/encryption. In the second selection, the ST author chooses the method used for encryption: Using one of the two AES-based Key Wrap methods specified in NIST SP 800-38F;Using one of the two the KTS-OAEP schemes for RSA as described in NIST SP 800-56B (KTSOAEP-basic described in section 9.2.3Using ECC CDH as described in NIST SP 800-56A section 6.2.2.2. The third selection should be made to reflect the key size. 2048/4096 is used for the RSA-based schemes, while the size of the prime modulus is used for ECC-based schemes. Support for 256-bit AES key sizes will be required for products entering evaluation after Quarter 3, 2015. Based on the method(s) selected, the last selection should be used to select the appropriate reference(s). The evaluator will examine the TSS to ensure there is a high-level description of how the key is protected and meets the appropriate specification.

Trusted Installation and Update for Add-ons

The evaluator will examine the TSS to verify that it states that the email client will reject add-ons from untrusted sources. The evaluator will examine the operational guidance to verify that it includes instructions on how to configure the email client with trusted add-on sources. The evaluator will perform the following tests:

- Test 1: The evaluator will create or obtain an add-on signed by a trusted source and attempt to install it. The evaluator verifies that the signature on the addon is valid and that the add-on can be installed.
- Test 2: The evaluator will create or obtain an add-on signed with an invalid certificate and attempt to install it. The evaluator verifies that the signed addon is rejected and cannot be installed.
- Test 3: The evaluator will create or obtain an add-on signed by a trusted source, modify the addon without resigning it, and attempt to install it. The evaluator verifies that the signed add-on is rejected and cannot be installed.

TSS
Guidance
Tests

Simple Authentication and Security Layer (SASL)

The evaluator will examine the TSS to determine that it describes the details of the email client connecting to a Mail Transfer Agent in terms of the SASL connection, along with email clientspecific options or procedures that might not be reflected in the specification. The evaluator will confirm that the operational guidance contains instructions for establishing the connection to the Mail Transfer Agent. The evaluator will also perform the following tests:

- Test 1: The evaluators shall ensure that the email client is able to initiate communications using POP, IMAP and SMTP and requiring SASL, setting up the connections as described in the operational guidance and ensuring that communication is successful.

- Test 2: The evaluator will ensure, for each communication channel with an authorized IT entity in tests 1, that a valid SASL handshake is performed. To perform this test, the evaluator will use a sniffer and a packet analyzer. The packet analyzer must indicate that the protocol in use is SASL.

TSS
Guidance
Tests

Key Combining

If keys are XORed together to form an intermediate key, the TSS section shall identify how this is performed (e.g., if there are ordering requirements, checks performed, etc.). The evaluator will also confirm that the TSS describes how the length of the output produced is at least the same as that of the data encryption key.
TSS

Key Wrapping

The evaluator will examine the TSS to ensure there is a high-level description of how the key is protected and meets the appropriate specification.
TSS

2.5 Evaluation Activities for Objective SFRs

The PP-Module does not define any objective requirements.

3 Evaluation Activities for SARs

The PP-Module does not define any SARs beyond those defined within the ApSo PP base to which it must claim conformance. It is important to note that a TOE that is evaluated against the PP-Module is inherently evaluated against this Base-PP as well. The ApSo PP includes a number of Evaluation Activities associated with both SFRs and SARs. Additionally, the PP-Module includes a number of SFR-based Evaluation Activities that similarly refine the SARs of the Base-PPs. The evaluation laboratory will evaluate the TOE against the Base-PP and supplement that evaluation with the necessary SFRs that are taken from the PP-Module.

4 Required Supplementary Information

This Supporting Document has no required supplementary information beyond the ST, operational guidance, and testing.

Appendix A - References

Identifier	Title
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and General Model, CCMB-2017-04-001, Version 3.1 Revision 5, April 2017. • Part 2: Security Functional Components, CCMB-2017-04-002, Version 3.1 Revision 5, April 2017. • Part 3: Security Assurance Components, CCMB-2017-04-003, Version 3.1 Revision 5, April 2017.
[AppPP]	Protection Profile for Application Software
[MS-OXCMAPIHTTP]	Messaging Application Programming Interface (MAPI) Extensions for HTTP