Application Software Extended Package for Email Clients

Version: 2.0
2015-6-18
National Information Assurance Partnership

Revision History

Version	Date	Comment
v 2.0	2015-06-18	Application Software Extended Package for Email Clients
v 1.0	2014-04-01	Initial release - Protection Profile for Email Clients

Contents

1. Introduction

1.1 Overview

Email clients are user applications that provide functionality to send, receive, access and manage email. The complexity of email content and email clients has grown over time. Modern email clients can render HTML as well as plaintext, and may include functionality to display common attachment formats, such as Adobe PDF and Microsoft Word documents. Some email clients allow their functionality to be modified by users through the addition of add-ons. Protocols have also been defined for communicating between email clients and servers. Some clients support multiple protocols for doing the same task, allowing them to be configured according to email server specifications.

The complexity and rich feature set of modern email clients make them a target for attackers, introducing security concerns. This document is intended to facilitate the improvement of email client security by requiring use of operating system security services, cryptographic standards, and environmental mitigations. Additionally, the requirements in this document define acceptable behavior for email clients regardless of the security features provided by the operating system.

This Extended Package (EP) along with the Protection Profile for Application Software ([AppPP]) provide a baseline set of Security Functional Requirements (SFRs) for email clients running on any operating system regardless of the composition of the underlying platform.

The terms email client and TOE are interchangeable in this document.

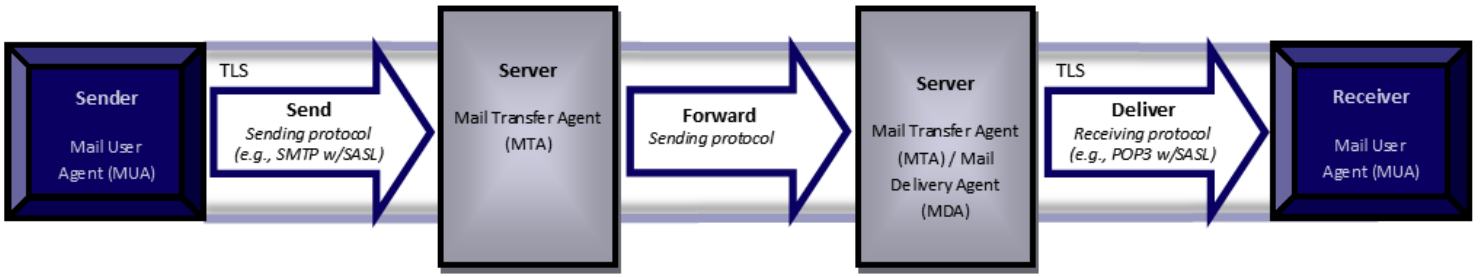

Figure 1 - Sending and Delivering Email over TLS

1.2 Terms

The following sections provide both Common Criteria and technology terms used in this Extended Package.

1.2.1 Common Criteria Terms

Common Criteria (CC)	Common Criteria for Information Technology Security Evaluation.
Extended Package (EP)	An implementation-independent set of security requirements for a category of products, which extends those in a Protection Profile.
Protection Profile (PP)	An implementation-independent set of security requirements for a category of products.
Security Target (ST)	A set of implementation-dependent security requirements for a specific product.
Target of Evaluation (TOE)	The product under evaluation. In this case, application software and its supporting documentation.
TOE Security Functionality (TSF)	The security functionality of the product under evaluation.
TOE Summary Specification (TSS)	A description of how a TOE satisfies the SFRs in a ST.
Security Functional Requirement (SFR)	A requirement for security enforcement by the TOE.
Security Assurance Requirement (SAR)	A requirement to assure the security of the TOE.

1.2.2 Technology Terms

ActiveSync	Microsoft protocol for synchronizing messaging and calendar data between mobile clients and email servers.
Add-on	Capabilities or functionality added to an application including plug-ins, extensions or other controls.
Email Client	Application used to send, receive, access and manage email provided by an email server. The terms email client and TOE are interchangeable in this document.
IMAP	Internet Message Access Protocol - Protocol for an email client to retrieve email from an email server over TCP/IP; IMAP4 defined in RFC 3501.
MAPI	Messaging Application Programming Interface - open specification used by email clients such as Microsoft Outlook and Thunderbird; defined in [MS-OXCMAPIHTTP].
POP	Post Office Protocol - Protocol for an email client to retrieve email from an email server over TCP/IP; POP3 defined in RFC 1939.
RPC	Remote Procedure Call - Protocol used by Microsoft Exchange to send/receive MAPI commands; defined in MS-OXCRPC.
S/MIME	Secure/Multipurpose Internet Mail Extensions - S/MIME is used to sign and/or encrypt messages at the request of the user upon sending email and to verify digital signature on a signed message upon receipt.
SMTP	Simple Mail Transfer Protocol - Protocol for an email client to send email to an email server over TCP/IP; SMTP defined in RFC 5321.

1.3 Compliant Targets of Evaluation

The Target of Evaluation (TOE) in this EP is an email client application running on a desktop or mobile operating system.

This EP describes the extended security functionality of email clients in terms of [CC]. As an extension of the App PP, it is expected that the content of this EP will be

appropriately combined with the App PP to include selection-based requirements in accordance with the selections and/or assignments made, and any optional and/or objective components to include: FCS_CKM.2.1, FCS_COP.1.1(*), FCS_RBG_EXT.2.*, FCS_TLSC_EXT.1.*, FIA_X509_EXT.1.*, FIA_X509_EXT.2.*.

An ST must identify the applicable version of the App PP and this EP in its conformance claims.

1.4 Use Cases

Email clients perform tasks associated primarily with the following use case.

[USE CASE 1] Sending, receiving, accessing, managing and displaying email

Email clients are used for sending, receiving, viewing, accessing, managing email in coordination with a mail server. Email clients can render HTML as well as plaintext, and can display common attachment formats.

2. Conformance Claims

Conformance Statement

The Protection Profile for Application Software ([AppPP]) defines the baseline Security Functional Requirements (SFRs) and Security Assurance Requirements (SARs) for application software products. This EP serves to extend the App PP baseline with additional SFRs and associated Assurance Activities specific to an email client. Assurance Activities are the actions that the evaluator performs in order to determine an email client's compliance to the SFRs.

This EP conforms to Common Criteria [CC] for Information Technology Security Evaluation, Version 3.1, Revision 4. It is CC Part 2 extended and CC Part 3 conformant. In order to be conformant to this EP, the ST must include all components in this EP and the associated App PP that are:

- unconditional (which are always required)
- selection-based (which are required when certain selections are chosen in the unconditional requirements)

and may include optional and/or objective components that are desirable but not required for conformance.

In accordance with CC Part 1, dependencies are not included when they are addressed by other SFRs. The assurance activities provide adequate proof that any dependencies are also satisfied.

3. Security Problem Description

The security problem is described in terms of the threats that the email client is expected to address, assumptions about the operational environment, and any organizational security policies that it is expected to enforce.

This Extended Package does not repeat the threats, assumptions, and organizational security policies identified in the App PP, though they all apply given the conformance and hence dependence of this EP on it. Together the threats, assumptions and organizational security policies of the App PP and those defined in this EP describe those addressed by an email client as the Target of Evaluation.

Notably, email clients are particularly at risk from the Network Attack threat identified in the App PP. Attackers can send malicious email messages directly to users, and the email client will render or otherwise process this untrusted content.

3.1 Threats

The following threat is specific to email clients, and represents an addition to those identified in the App PP.

T.FLAWED_ADDON
Email client functionality can be extended with integration of third-party utilities and tools. This expanded set of capabilities is made possible via the use of add-ons. The tight integration between the basic email client code and the new capabilities that add-ons provide increases the risk that malefactors could inject serious flaws into the email client application, either maliciously by an attacker, or accidentally by a developer. These flaws enable undesirable behaviors including, but not limited to, allowing unauthorized access to sensitive information in the email client, unauthorized access to the device's file system, or even privilege escalation that enables unauthorized access to other applications or the operating system.

4. Security Objectives

This Extended Package adds SFRs to objectives identified in the App PP and describes an additional objective specific to this EP.

4.1 Security Objectives for the TOE

O.MANAGEMENT
Addressed by: FDP_NOT_EXT.1, FDP_NOT_EXT.2, FMT_MOF_EXT.1

O.PROTECTED_STORAGE
Addressed by: FCS_CKM_EXT.3, FCS_CKM_EXT.4, FCS_CKM_EXT.5, FCS_COP_EXT.2, FCS_IVG_EXT.1, FCS_KYC_EXT.1, FCS_NOG_EXT.1, FCS_SAG_EXT.1, FCS_SMC_EXT.1

O.PROTECTED_COMMS
Addressed by: FCS_SMIME_EXT.1, FDP_SMIME_EXT.1, FIA_SASL_EXT.1, FIA_X509_EXT.3, FTP_ITC_EXT.1

O.ADDON_INTEGRITY
To address issues associated with malicious or flawed plug-ins or extensions, conformant email clients implement mechanisms to ensure their integrity. This includes verification at installation time and update.
Addressed by: FPT_AON_EXT.1, FPT_AON_EXT.2

5. Security Requirements

This chapter describes the security requirements which have to be fulfilled by the email client. Those requirements comprise functional components from Part 2 of [CC]. The following notations are used:

- **Selection** (denoted by *italicized text*): is used to select one or more options provided by the [CC] in stating a requirement.
- **Assignment** operation (denoted by *italicized text*): is used to assign a specific value to an unspecified parameter, such as the length of a password. Showing the value in square brackets indicates assignment.
- **Iteration** operation: are identified with a number inside parentheses (e.g. "(1)").

5.1 Security Functional Requirements

The Security Functional Requirements included in this section are derived from Part 2 of the Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, with additional extended functional components.

5.1.1 Cryptographic Support (FCS)

FCS_SMIME_EXT.1 Secure/Multipurpose Internet Mail Extensions (S/MIME)

FCS_SMIME_EXT.1.1

The email client shall implement both a sending and receiving S/MIME v3.2 Agent as defined in RFC 5751, using CMS as defined in RFCs 5652, 5754, and 3565.

Application Note: The RFCs allow for an agent to be either sending or receiving, or to include both capabilities. The intent of this requirement is to ensure that the email client is capable of both sending and receiving S/MIME v3.2 messages.

FCS_SMIME_EXT.1.2

The email client shall transmit the ContentEncryptionAlgorithmIdentifier for AES-128 CBC and AES-256 CBC as part of the S/MIME protocol.

Application Note: AES was added to CMS as defined in RFC 3565.

FCS_SMIME_EXT.1.3

The email client shall present the digestAlgorithm field with the following Message Digest Algorithm identifiers [**selection**: *id-sha256*, *id-sha384*, *id-sha512*] and no others as part of the S/MIME protocol.

FCS_SMIME_EXT.1.4

The email client shall present the AlgorithmIdentifier field with the following sha256withRSAEncryption and [**selection**:

sha384WithRSAEncryption,
sha512WithRSAEncryption,
ecdsa-with-SHA256,
ecdsa-with-sha384,
ecdsa-with-sha512

] and no other algorithms as part of the S/MIME protocol.

Application Note: RFC 5751 mandates that receiving and sending agents support RSA with SHA256. The algorithms to be tested in the evaluated configuration are limited to the algorithms specified in the FCS_SMIME_EXT.1.4 selection. Any other algorithms implemented that do not comply with these requirements should not be included in an evaluated email client.

FCS_SMIME_EXT.1.5

The email client shall support use of different private keys (and associated certificates) for signature and for encryption as part of the S/MIME protocol.

FCS_SMIME_EXT.1.6

The email client shall only accept a signature from a certificate with the digitalSignature bit set as part of the S/MIME protocol.

Application Note: It is acceptable to assume that the digitalSignature bit is set in cases where there is no keyUsage extension.

FCS_SMIME_EXT.1.7

The email client shall implement mechanisms to retrieve certificates and certificate revocation information [**selection**: *for each signed/encrypted message sent/received* , ***[assignment**: frequency]*] as part of the S/MIME protocol.

Application Note: In accordance with FIA_X509_EXT.1.1 [AppPP], certificate revocation may use Certificate Revocation List (CRL) or Online Certificate Status Protocol (OCSP). The email client can define how this mechanism behaves, including whether it utilizes the underlying OS, but it is required that a mechanism exist such that revocation status is supported and so that certificates can be retrieved for sending/receiving messages. Frequency is configurable in FMT_MOF_EXT.1.1. In this requirement, frequency can be interpreted as a one-time function with local storage, as a regularly scheduled retrieval, or as a mechanism that requires manual intervention. If the retrieval mechanism is periodic in nature, then the ST author will need to include an iteration of FCS for storage of revocation information; storage of certificates is covered in FCS_CKM. The import of certificates and certificate chains is not included in this requirement, but is covered in FIA_X509 and FMT_MOF.

Assurance Activity

TSS

The evaluator shall verify that the version of S/MIME implemented by the email client is present in the TSS. The evaluator shall also verify that the algorithms supported are specified, and that the algorithms specified are those listed for this component.

The evaluator shall verify that the TSS describes the ContentEncryptionAlgorithmIdentifier and whether the required behavior is performed by default or may be configured.

The evaluator shall verify that the TSS describes the digestAlgorithm and whether the required behavior is performed by default or may be configured.

The evaluator shall verify that the TSS describes the AlgorithmIdentifier and whether the required behavior is performed by default or may be configured.

The evaluator shall verify that the TSS describes the retrieval mechanisms for both certificates and certificate revocation as well as the frequency at which these mechanisms are implemented.

Guidance

The evaluator shall also review 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 shall verify 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 shall verify 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 shall verify 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 shall verify that the AGD guidance includes the configuration of these mechanisms.

Tests

The evaluator shall perform the following tests:

These tests can be performed in conjunction with the tests defined in FIA_X509 for certificate/certificate chain verification and FDP_NOT_EXT.1.

- ***Test 1:** The evaluator shall both send and receive 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 shall both send and receive 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 shall send a signed message using each of the algorithms specified in the ST and use a man-in-the-middle 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 shall verify that the received message fails the signature validation check.*
- ***Test 3:** The evaluator shall both send and receive an encrypted message using each of the algorithms specified in the ST. The evaluator shall verify that the contents are encrypted in transit and that the received message decrypts. After verifying the message decrypts, the evaluator shall send an encrypted message using each of the algorithms specified in the ST and use a man-in-the-middle tool to modify at least one byte of the message such that the encryption is no*

longer valid. The evaluator shall verify that the received message fails to decrypt.

- ***Test 4:*** *The evaluator shall both send and receive a message that is both signed and encrypted. In addition, the evaluator shall use 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 shall verify that the received message fails to decrypt, fails the signature validation check, and/or both.*
- ***Test 5:*** *The evaluator shall send a signed message to the email client using a signature algorithm not supported according to the digestAlgorithm ID (e.g., SHA1). The evaluator shall verify that the email client provides a notification that the contents cannot be verified because the signature algorithm is not supported.*
- ***Test 6:*** *The evaluator shall send an encrypted message to the email client using an encryption algorithm not supported according to the AlgorithmIdentifier field. The evaluator shall verify that the email client does not display/decrypt the contents of the message.*
- ***Test 7:*** *The evaluator shall send the email client a message signed by a certificate without the digitalSignature bit set. The evaluator shall verify that the email client notifies the user that the signature is invalid.*
- ***Test 8:*** *The evaluator shall send the email client a message signed by a certificate without the Email Protection purpose in the extendedKeyUsage. The evaluator shall verify that the email client notifies the user that the signature is invalid.*
- ***Test 9:*** *The evaluator shall verify that the email client uses OCSP or downloads the CRL at the assigned frequency.*

FCS_CKM_EXT.3 Protection of Key and Key Material

FCS_CKM_EXT.3.1

The email client shall [**selection**: *not store keys in non-volatile memory*, *only store keys in non-volatile memory when wrapped as specified in FCS_COP_EXT.2 unless the key meets any one of following criteria: [**selection**:*

The plaintext key is not part of the key chain as specified in FCS_KYC_EXT.1 ,

The plaintext key will no longer provide access to the encrypted data after initial provisioning ,

*The plaintext key is a key split that is combined as specified in FCS_SMC_EXT.1, and the other half of the key split is either [**selection**: wrapped as specified in FCS_COP_EXT.2 , derived and not stored in non-volatile memory] ,*

The plaintext key is stored on an external storage device for use as an authorization factor ,

The plaintext key is used to wrap a key as specified in FCS_COP_EXT.2 that is already wrapped as specified in FCS_COP_EXT.2 ,

The plaintext key is the public portion of the key pair

]] .

Application Note: The plaintext key storage in non-volatile memory is allowed for several reasons. If the keys exist within protected memory that is not user accessible on the email client or operational environment, the only methods that allow it to play a security relevant role is if it is a key split or providing additional layers of wrapping or encryption on keys that have already been protected.

Assurance Activity

TSS

The evaluator shall verify the TSS for a high level description of method used to protect keys stored in non-volatile memory.

The evaluator shall verify the TSS to ensure it describes the storage location of all keys and the protection of all keys stored in non-volatile 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 non-volatile memory and plaintext keys in non-volatile memory meet one of the criteria for storage.

Guidance

N/A

FCS_CKM_EXT.4 Cryptographic Key Destruction

FCS_CKM_EXT.4.1

The email client shall [**selection**:

invoke platform-provided key destruction,

*implement key destruction using [**selection**:*

*For volatile memory, the erasure shall be executed by a single direct overwrite [**selection**:*

consisting of a pseudo-random pattern using the email client's RBG,

consisting of a pseudo-random pattern using the host platform's RBG,

consisting of zeroes

] ,

*For non-volatile storage, the erasure shall be executed by [**selection**:*

single,

three or more times

*] overwrite of key data storage location consisting of [**selection**:*

a pseudo random pattern using the email client's RBG (as specified in FCS_RBG_EXT.1 [AppPP],

a pseudo-random pattern using the host platform's ,

a static pattern

]

]

] that meet the following: [**selection**:

NIST SP800-88,

no standard

] for destroying all keying material and cryptographic security parameters when no longer needed.

Application Note: For the purposes of this requirement, keying material refers to authentication data, passwords, symmetric keys, data used to derive keys, etc.

The destruction indicated above applies to each intermediate storage area for key/cryptographic critical security parameters (i.e., any storage, such as memory buffers, that is included in the path of such data) upon the transfer of the key/cryptographic critical security parameter to another memory location.

Assurance Activity

TSS

If the platform provides the key destruction, then the evaluator shall examine the TSS to verify that it describes how the key destruction functionality is invoked.

If the application invokes key destruction, the evaluator shall check 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 shall check 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").

Guidance

N/A

Tests

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 non-volatile memory.*

FCS_KYC_EXT.1 Key Chaining

FCS_KYC_EXT.1.1

The email client shall maintain a key chain of: [**selection**:
one,
a key stored in platform key storage,
*intermediate keys originating from: [**selection**:*
a password as specified in FCS_CKM_EXT.5.1,
one or more other authorization factor(s),
credentials stored in platform key storage
*] to the data encryption/decryption key(s) using the following method(s): [**selection**:*
utilization of the platform key storage,
utilization of platform key storage that performs key wrap with a TSF provided key,
implement key wrapping as specified in FCS_COP_EXT.2,
implement key combining as specified in FCS_SMC_EXT.1
*] while maintaining an effective strength of [**selection**:*
128 bits,
256 bits
]
]

Application Note: Key Chaining is the method of using multiple layers of encryption keys to ultimately secure the data encryption key. The number of intermediate keys will vary. This applies to all keys that contribute to the ultimate wrapping or derivation of the data encryption key; including those in protected areas. This requirement also describes how keys are stored.

Assurance Activity

TSS

The evaluator shall verify 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 shall 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 meet FCS_COP_EXT.2. The evaluator shall verify 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 shall 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.*

If the platform provides the IV generation, then the evaluator shall examine the TSS to verify that it describes how the IV generation is invoked.

Guidance

N/A

FCS_IVG_EXT.1 Initialization Vector Generation

FCS_IVG_EXT.1.1

The email client shall create IVs in the following manner: [**selection**:

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.

]

Application Note: 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.

Assurance Activity

TSS

The evaluator shall ensure the TSS describes how IVs and tweaks are handled (based on the AES mode). The evaluator shall confirm that the IVs and tweaks meet the stated requirements.

Guidance

N/A

5.1.2 User Data Protection (FDP)

FDP_NOT_EXT.1 Notification of S/MIME Status

FDP_NOT_EXT.1.1

The email client shall display a notification of the S/MIME status of received emails upon viewing.

Application Note: S/MIME status is whether the email has been signed or encrypted and whether the signature verifies and the

associated certificate validates. This notification must at least display when the email content is viewed. Many implementations also display the S/MIME status of each email when all emails are viewed as a list.

Assurance Activity

TSS

The evaluator shall ensure that the TSS describes notifications of S/MIME status, including whether S/MIME status is also indicated upon viewing a list of emails.

Guidance

The evaluator shall verify 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.

Tests

The evaluator shall perform the following tests and may perform them in conjunction with the tests for FCS_SMIME_EXT.1:

- ***Test 1:*** *The evaluator shall send the client an unencrypted and unsigned email and verify that no notifications are present upon viewing.*
- ***Test 2:*** *The evaluator shall send the client an encrypted email and verify that the encrypted notification is present upon viewing.*
- ***Test 3:*** *The evaluator shall send the client a valid signed email and verify that the signed notification is present upon viewing.*
- ***Test 4:*** *The evaluator shall 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.*

FDP_SMIME_EXT.1 S/MIME

FDP_SMIME_EXT.1.1

The email client shall use S/MIME to sign, verify, encrypt, and decrypt mail.

Application Note: Note that this requirement does not mandate that S/MIME be used for all incoming/outgoing messages, or that the email client automatically encrypt and/or sign/verify all sent or received messages. This requirement only specifies that the mechanism for digital signature and encryption must be S/MIME.

Assurance Activity

TSS

The evaluator shall verify 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 shall verify 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.

Guidance

The evaluator shall ensure that the AGD guidance includes instructions for configuring a certificate for S/MIME use and

instructions for signing and encrypting email.

Tests

Tests for this element are performed in conjunction with tests for FCS_SMIME_EXT.1 and FDP_NOT_EXT.1

5.1.3 Identification and Authentication (FIA)

FIA_X509_EXT.3 X509 Authentication and Encryption

FIA_X509_EXT.3.1

The email client shall use X.509v3 certificates as defined by RFC 5280 to support encryption and authentication for S/MIME.

FIA_X509_EXT.3.2

The email client shall prevent the establishment of a trusted communication channel when the peer certificate is deemed invalid.

Application Note: Trusted communication channels include any of TLS performed by the email client. Validity is determined by the certificate path, the expiration date, and the revocation status in accordance with RFC 5280.

FIA_X509_EXT.3.3

The email client shall prevent the installation of code if the code signing certificate is deemed invalid.

FIA_X509_EXT.3.4

The email client shall prevent the encryption of email if the email protection certificate is deemed invalid.

FIA_X509_EXT.3.5

The email client shall prevent the signing of email if the email protection certificate is deemed invalid.

Assurance Activity

TSS

The evaluator shall check 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 shall 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.

Guidance

The evaluator shall verify that the administrative guidance contains any necessary instructions for configuring the operating environment so that the email client can use the certificates.

Tests

The evaluator shall perform the following tests:

- ***Test 1:*** *The evaluator shall perform Test 1 for each function listed in FIA_X509_EXT.2.1 ([AppPP]) that requires the use of certificates. The evaluator shall demonstrate that using a certificate without a valid certification path results in the function failing. The evaluator shall 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 shall demonstrate that using a valid certificate that requires certificate validation checking to be performed in at least some part by communicating with a non-TOE IT entity. The evaluator shall 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 ([AppPP]) is performed. If the selected action is administrator-configurable, then the evaluator shall follow the operational guidance to determine that all supported administrator-configurable options behave in their documented manner.*

5.1.4 Security Management (FMT)

FMT_MOF_EXT.1 Management of Functions Behavior

FMT_MOF_EXT.1.1

The email client shall be capable of performing the following management functions, controlled by the user or administrator as shown:

- X: Mandatory
- O: Optional

Management Function	Administrator	User
Enable/disable downloading embedded objects globally and by [**selection**: *domain*, *sender*, *no other method*]	O	O
Enable/disable plaintext only mode globally and by [**selection**: *domain*, *sender*, *no other method*]	O	O
Enable/disable rendering and execution of attachments globally and by [**selection**: *domain*, *sender*, *no other method*]	O	O
Enable/disable email notifications	O	O
Configure a certificate repository for encryption	O	O
Configure whether to establish a trusted channel or disallow establishment if the email client cannot establish a connection to determine the validity of a certificate	O	O
Configure message sending/receiving to only use cryptographic algorithms defined in FCS_SMIME_EXT.1	O	O
Configure CRL retrieval frequency	O	O
Enable/disable support for add-ons	O	O
Change password/passphrase authentication credential	O	O
Disable key recovery functionality	O	O
Configure cryptographic functionality	O	O
[**assignment**: Other management functions]	O	O

Application Note: For these management functions, the term "Administrator" refers to the administrator of a non-mobile device or the device owner of a mobile device. The Administrator is responsible for management activities, including setting the policy that is applied by the enterprise on the email client. The Administrator could be acting remotely and could be the MTA

administrator acting through a centralized management console or dashboard. Applications used to configure enterprise policy should have their own identification and authorization and additional security requirements to ensure that the remote administration is trusted.

The intent of this requirement is to allow the Administrator to configure the email client with a policy that may not be over-ridden by the user. If the Administrator has not set a policy for a particular function, the user may still perform that function. Enforcement of the policy is done by the email client itself, or the email client and the email client platform in coordination with each other.

The function to configure whether to establish a trusted channel corresponds to the functionality described in FIA_X509_EXT.2.2 ([AppPP]). The Administrator has the option of accepting or rejecting all certificates that cannot be validated, accepting a given certificate that cannot be validated, or not accepting a given certificate that cannot be validated. Depending on the choice that the Administrator has made in FIA_X509_EXT.2.2 ([AppPP]), the trusted connection will either be allowed for all certificates that cannot be validated, disallowed for all certificates that cannot be validated, allowed for a given certificate that cannot be validated, or disallowed for a given certificate that cannot be validated.

If password or passphrase authorization factors are implemented by the email client, then the appropriate "change" selection must be included.

If the email client provides configurability of the cryptographic functions (for example, key size), even if the configuration is the form of parameters that may be passed to cryptographic functionality implement on the email client platform, then "configure cryptographic functionality" will be included, and the specifics of the functionality offered can either be written in this requirement as bullet points, or included in the TSS.

If the email client does include a key recovery function, the email client must provide the capability for the user to turn this functionality off so that no recovery key is generated and no keys are permitted to be exported.

Assurance Activity

The assurance 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 assurance activity does not need to be performed.

TSS

The evaluator shall verify that the TSS describes those management functions which may only be configured by the email client platform administrator and cannot be over-ridden by the user when set according to policy.

Change Password: The evaluator shall 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 shall determine from the TSS for other requirements (FCS_*) what portions of the cryptographic functionality are configurable.

Guidance

The evaluator shall 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 shall review the AGD documentation to determine that there are instructions

for manipulating all of the claimed mechanisms.

Tests

The evaluator shall perform the following tests:

- ***Test 1:*** *The evaluator shall verify that functions perform as intended by enabling, disabling, and configuring the functions.*
- ***Test 2:*** *The evaluator shall set management functions which are controlled by the (enterprise) administrator and cannot be overridden by the user. The evaluator shall 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 shall devise a test that ensures that the key recovery capability has been or can be disabled following the guidance provided by the vendor.*

5.1.5 Protection of the TSF (FPT)

FPT_AON_EXT.1 Support for Only Trusted Add-ons

FPT_AON_EXT.1.1

The email client shall include the capability to load [**selection**: *trusted add-ons*, *no add-ons*] .

Application Note: FPT_AON_EXT.2 depends upon the selection made here. If the email client does not include support for installing only trusted add-ons, this requirement can be met by demonstrating the ability to disable all support for add-ons as specified in FMT_MOF_EXT.1. Cryptographic verification (i.e., trust) of add-ons is tested in FPT_AON_EXT.2.1

Assurance Activity

TSS

The evaluator shall verify that the TSS describes whether the email client is capable of loading trusted add-ons.

Guidance

The evaluator shall examine the operational guidance to verify that it includes instructions on loading trusted add-on sources.

Tests

The evaluator shall perform the following test:

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

5.1.6 Trusted Path/Channels (FTP)

FTP_ITC_EXT.1 Inter-TSF Trusted Channel

FTP_ITC_EXT.1.1

The email client shall initiate or receive communication via the trusted channel.

FTP_ITC_EXT.1.2

The email client shall communicate via the trusted channel for [**selection**:

IMAP,
SMTP,
POP,
MAPI Extensions for HTTP,
MAPI/RPC,
ActiveSync,
*[**assignment**: other protocol (reference RFC or specification)]*

] .

Application Note: FIA_SASL_EXT.1 depends upon the selection(s) made here. For example, if *POP* is chosen, then FIA_SASL_EXT.1.2 shall be included in the ST. Selections must include at least one sending and one receiving protocol. If the assignment is used, the ST author must also include a reference for the protocol (e.g., an RFC number).

Assurance Activity

TSS

The evaluator shall 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 [AppPP], along with email client-specific options or procedures that might not be reflected in the specification.

Guidance

The evaluator shall confirm that the operational guidance contains instructions for establishing the connection to the Mail Transfer Agent.

Tests

The evaluator shall 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 shall 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 shall use a sniffer and a packet analyzer. The packet analyzer must indicate that the protocol in use is TLS.*

A. Optional Requirements

The baseline requirements (those that must be performed by the TOE) are contained in the body of this EP. Additionally, there are three other types of requirements specified in Appendix A, Appendix B, and Appendix C. The first type (in this Appendix) are requirements that can be included in the ST, but are not required in order for an email client to claim conformance to this EP. The second type (in Appendix B) are requirements based on selections in the body of the EP: if certain selections are made, then additional requirements in that appendix must be included. The third type (in Appendix C) are components that are not required in order to conform to this EP, but will be included in the

baseline requirements in future versions of this EP, so adoption by vendors is encouraged. Note that the ST author is responsible for ensuring that requirements that may be associated with those in Appendix A, Appendix B, and Appendix C but are not listed (e.g., FMT-type requirements) are also included in the ST.

FCS_CKM_EXT.5 Cryptographic Key Derivation (Password/Passphrase Conditioning)

FCS_CKM_EXT.5.1

A password/passphrase used to generate a password authorization factor shall enable up to [**selection**:

*[**assignment**: positive integer of 64 or more] characters in the set of [**selection**:*

upper case characters,

lower case characters,

numbers,

*special characters: !, @, #, $, %, ^, &, *, (,),*

*[**assignment**: other supported special characters]*

]

] and shall perform [Password-based Key Derivation Functions] in accordance with a specified cryptographic algorithm [**selection**:

HMAC-SHA-256,

HMAC-SHA-384,

HMAC-SHA-512

] with [**selection**: *[**assignment**: positive integer of 4096 or more]*] iterations, and output cryptographic key sizes of [**selection**:

128 bits,

256 bits

] that meet NIST SP 800-132.

Application Note: 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 ST Author. SP 800-132 requires the use of a pseudo-random function (PRF) consisting of HMAC with an approved hash function. The ST 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 EP, 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.

Assurance Activity

TSS

The evaluators shall check the TSS section to determine that it specifies that a capability exists to accept passwords/passphrases with the minimum number of characters specified in the ST in this assignment statement.

The evaluator shall examine 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 shall check 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 shall verify that these are supported by the selections in this component as well as the selections concerning the hash function itself. The evaluator shall verify 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 800-132-based conditioning of the password/passphrase, the required assurance activities will be performed when doing the assurance activities for the appropriate requirements (FCS_COP.1.1(4) [AppPP]). 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.

Guidance

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.

Tests

The evaluator shall 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 shall verify that the iteration count for PBKDFs performed by the email client comply with NIST SP 800-132 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 password-based key derivation (including but not limited to increasing the iteration count).

FCS_SAG_EXT.1 Cryptographic Salt Generation

FCS_SAG_EXT.1.1

The email client shall only use salts that are generated by a [**selection**:

RNG as specified in FCS_RBG_EXT.1,

RNG provided by the host platform

]

Application Note: The salt must be random.

Assurance Activity

TSS

The evaluator shall ensure the TSS describes how salts are generated. The evaluator shall confirm that the salt is generated using an described in FCS_RBG_EXT.1 [AppPP] 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 shall verify 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.

Guidance

N/A

FCS_NOG_EXT.1 Cryptographic Nonce Generation

FCS_NOG_EXT.1.1

The email client shall only use unique nonces with a minimum size of [64] bits.

Application Note: Nonces must be unique.

Assurance Activity

TSS

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

Guidance

N/A

FDP_NOT_EXT.2 Notification of URI

FDP_NOT_EXT.2.1

The email client shall display the full Uniform Resource Identifier (URI) of any embedded links.

Application Note: 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.

Assurance Activity

TSS

The evaluator shall verify that the TSS includes a description of how embedded links are rendered and the method by which the URI of the link is displayed.

Guidance

The evaluator shall ensure that the AGD guidance includes instructions (with any appropriate visual figures) for viewing the URI of an embedded link.

Tests

The evaluator shall perform the following test:

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

FDP_PST_EXT.1 Storage of Persistent Information

FDP_PST_EXT.1.1

The email client shall be capable of operating without storing persistent information to the client platform with the following exceptions: [**selection**: *credential information*, *administrator provided configuration information*, *certificate revocation information*, *no exceptions*] .

Application Note: 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.

Assurance Activity

TSS

The evaluator shall 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 shall confirm that the persistent data described is limited to the data identified in the selection.

Guidance

N/A

Tests

The evaluator shall perform the following tests:

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

FDP_REN_EXT.1 Rendering of Message Content

FDP_REN_EXT.1.1

The email client shall have a plaintext-only mode which disables the rendering and execution of [**selection**:

HTML,

JavaScript,

*[**assignment**: other embedded content types]* ,

no embedded content types

] .

Application Note: Plaintext only mode prevents the automatic downloading, rendering and execution of images, external resources and embedded objects such as HTML or JavaScript objects. FMT_MOF_EXT.1.1 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.

Assurance Activity

TSS

The evaluator shall ensure that the TSS describes plaintext only mode for sending and receiving messages. The evaluator shall verify 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 shall ensure that the description indicates that embedded objects of these types are not rendered or executed and images/external resources are not automatically downloaded.

Guidance

The evaluator shall examine the AGD guidance and verify that it contains instructions for enabling plaintext only mode.

Tests

The evaluator shall perform the following tests:

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

B. Selection-Based Requirements

As indicated in the introduction to this EP, the baseline requirements (those that must be performed by the email client or its underlying platform) are contained in the App PP and in the body of this EP. There are additional requirements based on selections from the App PP and/or in the body of the EP: if certain selections are made, then additional requirements below will need to be included.

FCS_COP_EXT.2 Key Wrapping

FCS_COP_EXT.2.1

The email client shall [**selection**:
use platform-provided functionality to perform Key Wrapping,
implement functionality to perform Key Wrapping
] in accordance with a specified cryptographic algorithm [**selection**:
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 [**selection**:
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: [**selection**:
"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)

] .

This requirement depends upon selection in FCS_KYC_EXT.1.1, FCS_CKM_EXT.3.1.

Application Note: 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 9.2.3
- Using 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).

Assurance Activity

TSS

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

Guidance

N/A

FCS_SMC_EXT.1 Key Combining

FCS_SMC_EXT.1.1

The email client shall combine submasks using the following method [**selection**:

exclusive OR (XOR),

SHA-256,

SHA-512

] to generate another key.

This requirement depends upon selection in FCS_KYC_EXT.1.1, FCS_CKM_EXT.3.1.

Application Note: This requirement specifies the way that a product may combine the various submasks by using either an XOR or an approved SHA-hash.

Assurance Activity

TSS

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 shall 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.

Guidance

N/A

FIA_SASL_EXT.1 Simple Authentication and Security Layer (SASL)

FIA_SASL_EXT.1.1

The email client shall implement support for Simple Authentication and Security Layer (SASL) that complies with RFC 4422.

This requirement depends upon selection in FTP_ITC_EXT.1.2.

Application Note: 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.

FIA_SASL_EXT.1.2

The email client shall support the POP3 CAPA and AUTH extensions for the SASL mechanism.

This requirement depends upon selection in FTP_ITC_EXT.1.2.

FIA_SASL_EXT.1.3

The email client shall support the IMAP CAPABILITY and AUTHENTICATE extensions for the SASL mechanism.

This requirement depends upon selection in FTP_ITC_EXT.1.2.

FIA_SASL_EXT.1.4

The email client shall support the SMTP AUTH extension for the SASL mechanism.

This requirement depends upon selection in FTP_ITC_EXT.1.2.

Application Note: 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.

Assurance Activity

TSS

The evaluator shall 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 client-specific options or procedures that might not be reflected in the specification.

Guidance

The evaluator shall confirm that the operational guidance contains instructions for establishing the connection to the Mail Transfer Agent.

Tests

The evaluator shall 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 shall 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 shall use a sniffer and a packet analyzer. The packet analyzer must indicate that the protocol in use is SASL.*

FPT_AON_EXT.2 Trusted Installation and Update for Add-ons

FPT_AON_EXT.2.1

The email client shall [**selection**: *provide the ability*, *leverage the platform*] to provide a means to cryptographically verify add-ons using a digital signature mechanism and [**selection**: *published hash*, *no other functions*] prior to installation and update.

This requirement depends upon selection in FPT_AON_EXT.1.1.

FPT_AON_EXT.2.2

The email client shall [**selection**: *provide the ability*, *leverage the platform*] to query the current version of the add-on.

This requirement depends upon selection in FPT_AON_EXT.1.1.

FPT_AON_EXT.2.3

The email client shall prevent the automatic installation of add-ons.

This requirement depends upon selection in FPT_AON_EXT.1.1.

Assurance Activity

TSS

The evaluator shall examine the TSS to verify that it states that the email client will reject add-ons from untrusted sources.

Guidance

The evaluator shall examine the operational guidance to verify that it includes instructions on how to configure the email client with trusted add-on sources.

Tests

The evaluator shall perform the following tests:

- ***Test 1:** The evaluator shall create or obtain an add-on signed by a trusted source and attempt to install it. The*

evaluator shall verify that the signature on the add-on is valid and that the add-on can be installed.

- ***Test 2:*** *The evaluator shall create or obtain an add-on signed with an invalid certificate and attempt to install it. The evaluator shall verify that the signed add-on is rejected and cannot be installed.*
- ***Test 3:*** *The evaluator shall create or obtain an add-on signed by a trusted source, modify the add-on without re-signing it, and attempt to install it. The evaluator shall verify that the signed add-on is rejected and cannot be installed.*

C. Objective Requirements

This Annex includes requirements that specify security functionality which also addresses threats. The requirements are not currently mandated in the body of this EP as they describe security functionality not yet widely-available in commercial technology. However, these requirements may be included in the ST such that the email client is still conformant to this EP, and it is expected that they be included as soon as possible.

None.

D. References

Identifier	Title
[CC]	Common Criteria for Information Technology Security Evaluation - • Part 1: Introduction and General Model, CCMB-2012-09-001, Version 3.1 Revision 4, September 2012. • Part 2: Security Functional Components, CCMB-2012-09-002, Version 3.1 Revision 4, September 2012. • Part 3: Security Assurance Components, CCMB-2012-09-003, Version 3.1 Revision 4, September 2012.
[AppPP]	Protection Profile for Application Software
[MS-OXCMAPIHTTP]	Messaging Application Programming Interface (MAPI) Extensions for HTTP

E. Acronyms

Acronym	Meaning
AES	Advanced Encryption Standard
CBC	Cipher Block Chaining
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List

CSP	Critical Security Parameter
DRBG	Deterministic Random Bit Generator
ECDSA	Elliptic Curve Digital Signature Algorithm
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
IETF	Internet Engineering Task Force
IV	Initialization Vector
MTA	Mail Transfer Agent
NIST	National Institute of Standards and Technology
PBKDF	Password-Based Key Derivation Function
PDF	Portable Document Format
RBG	Random Bit Generator
RFC	Request for Comment (IETF)
RSA	Rivest Shamir Adelman
TLS	Transport Layer Security