# TEST ASSERTIONS

# eSENS CONFORMANCE PROFILE

**Document Reviewers**

| Name | Role |
|---|---|
| Pim van der Eijk | |
| Muhammet Yildiz | |
| Yang Chen | |
| Melis Ozgur Cetinkaya Demir | |
| David Hixon | |
| Maarten Daniels | |
| Christian Koch | |

**Document History**

| Revision | Date | Description | Modified by |
|---|---|---|---|
| Draft | 09/04/2015 | Document creation | |
| Draft2 | 08/05/2015 | Update after reviewers comments | |
| Draft3 | 02/06/2015 | Update after reviewers comments | |
| Draft4 | 12/06/2015 | Update after reviewers comments | |
| Draft5 | 14/07/2015 | Update after reviewers comments | |
| Draft6 | 20/08/2015 | Update after reviewers comments | |

**Introduction**

This document describes the test assertions for the e-SENS Message exchange protocol which is realized as a profile of the ebMS3 and AS4 standards. The e-SENS AS4 feature set is, with some exceptions, a subset of the feature set of the AS4 ebHandler Conformance Profile.

The purpose of the test assertions is to focus more on what is expected from the implementation rather than how to test it. In fact, some of the test assertions might not be testable (transformed into test cases).

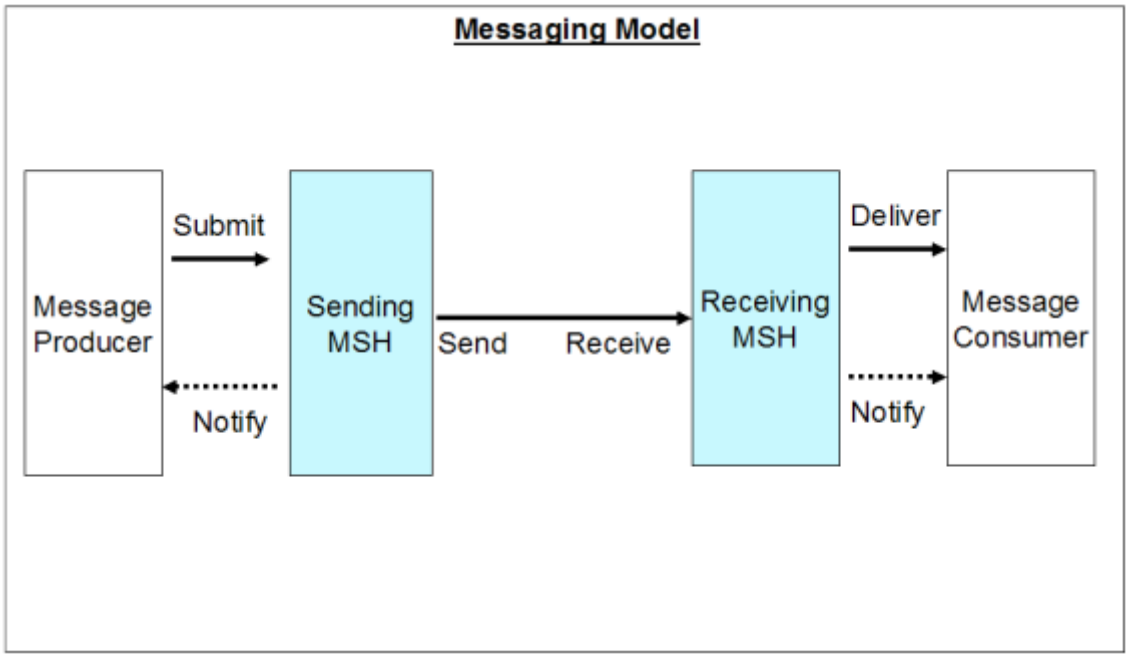
**System overview**

System entities

- Message Service Handler (MSH): is an entity that is able to generate or process eSENS AS4 messages.
- Producer: it is an entity (e.g. application) that interacts with a sending MSH to initiate the sending of a user message.
- Consumer: it is an entity that interacts with a receiving MSH to consume data from a received user message.

Messaging Model

Business applications or middleware, acting as Producer, Submit message content and metadata to the Sending MSH, which packages this content and sends it to the Receiving MSH of the business partner, which in turn Delivers the message to another business application that Consumes the message content and metadata. Subject to configuration, Sending and Receiving MSH may Notify Producer or Consumer of particular events.



Definitions

- User Message: it is a message that contains a User Message unit. It allows transmitting data interpreted by a Consumer.

- Signal Message: it is a message that contains a Signal Message unit. It allows transmitting data interpreted by an MSH (as a signal).

- MEP (Message Exchange Pattern): It is an agreement between sending and receiving MSHs. Some aspects of MEPs supported in the messaging layer include:

    - Specifying the correlation between messages sent and received in the message header.

    - Message binding to the underlying transfer-protocol.

Two MEPs are defined in this specification, not exclusive of others:

• The One-Way MEP which governs the exchange of a single User Message Unit unrelated to other User Messages. Its label is "oneway".
• The Two-Way MEP which governs the exchange of two User Message Units in opposite directions, the first one to occur is labeled "request", the other one "reply". In an actual instance, the "reply" must reference the "request" using eb:RefToMessageId.

- PMode (Processing Mode): A PMode is the contextual information that governs the processing of a particular message (thus is basically a set of configuration parameters).

Abbreviation

- SMSH: an MSH in the sending role.

- RMSH: an MSH in the receiving role.

Notes

- In this document, there are 2 lists of test assertions

    - A list for specific to the e-SENS profile requirements. Those test assertions are labelled eSENS_TAXX.

    - A list for generic AS4 requirements. Those test assertions are labelled AS4_TAXX.

- Test assertions not related to the AS4 protocol (load, size, volume…) are provided in a separate document (details still need to be discussed to have a normative source).

- For the details on the MSHs configuration, please refer to the document "PMode_Parameters_e-SENS_profile.doc". Configurations as described in test assertions:

| Configuration in predicate | PMode parameters |
|---|---|
| "*SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile*" | PModes are set according to the document "PMode_Parameters_e-SENS_profile.doc". |
| "*SMSH and RMSH are configured to exchange AS4 messages: One-Way/Push MEP*" | - PMode[1].MEP: set to One-Way.<br>- PMode[1].MEPBinding: set to Push. |
| "*SMSH and RMSH are configured to exchange AS4 messages: Two-Way/Push-and-Push MEP*" | - PMode[1].MEP: set to Two-Way.<br>- PMode[1].MEPBinding: set to Push-and-Push. |

- In order to cover some requirement, MSHs are sometimes misconfigured or "simulated" to produce AS4 messages not conform to the eSENS profile. This can also be achieved by intercepting the messages and altering them before they reach their destination.

Links

- [e-SENS] (specifications source): http://wiki.ds.unipi.gr/display/ESENS/ABB+-+Message+Exchange+Protocol

- [EBMS3]: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/core/os/

- [AS4]: http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/profiles/AS4-profile/v1.0/os/AS4-profile-v1.0-os.html

- [XML10]: http://www.w3.org/TR/REC-xml/

**e-SENS specific test assertions**

**TA id**:

eSENS_TA01

**Normative source**:

[e-SENS]

"*The AS4 ebHandler Conformance Profile is the AS4 conformance profile that provides support for Sending and Receiving roles using Push channel bindings. Support is required for the following Message Exchange Patterns:*

- *One Way / Push*
- *Two Way / Push-and-Push*"

**Target**:

"One-Way/Push" MEP

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- SMSH sends an AS4 User message to the RMSH.

**Predicate**:

The RMSH returns a non-repudiation receipt within a HTTP response with status code 2XX (for more details on http response codes please refer to https://issues.oasis-open.org/browse/EBXMLMSG-57?jql=project%20%3D%20EBXMLMSG).

**Prescription Level**:

Mandatory

**Tag**:

Message Exchange pattern, One-Way/Push

**TA id**:

eSENS_TA02

**Normative source**:

[e-SENS]

" *Message Exchange Patterns:*

- *One Way / Push*
- *Two Way / Push-and-Push*"

And

"*AS4 provides multiple mechanisms to correlate messages within a particular flow.*

*1.UserMessage/MessageInfo/RefToMessageId provides a way to express that a message is a response to a single specific previous message. Presence of a RefToMessageId is required in response messages in Two Way message exchanges. By default, exchanges are considered One Way."*

**Target**:

"Two-Way/Push-and-Push" MEP, CollaborationInfo

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: Two-Way/Push-and-Push MEP.
- SMSH sends an AS4 User Message (M1 with ID MessageId) that requires a consumer response to the RMSH.

**Predicate**:

The RMSH sends back a User Message (M2) with element REFTOMESSAGEID set to MESSAGEID (of M1).

**Prescription Level**:

Mandatory

**Tag**:

Message Exchange pattern, Two-Way/Push-and-Push, correlation Request/Response

**Variable**:

MESSAGEID: XML element (M1)Messaging/UserMessage/MessageInfo/MessageId

REFTOMESSAGEID: XML element (M2)Messaging/UserMessage/MessageInfo/ RefToMessageId

**TA id**:

eSENS_TA03

**Normative source**:

[e-SENS]

"*Both UserMessage/PartyInfo/From and UserMessage/PartyInfo/To must not include more than one PartyId element*"

**Target**:

User Message single exchange parties

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- SMSH and RMSH exchange several AS4 User Messages.

**Predicate**:

Each exchanged AS4 message contains single ORIGIN and DESTINATION elements.

**Prescription Level**:

Mandatory

**Tag**:

User Message, party info

**Variable**:

ORIGIN: XML element Messaging/UserMessage/PartyInfo/From

DESTINATION: XML element Messaging/UserMessage/PartyInfo/To

**TA id**:

eSENS_TA04

*Note:*

*This test assertion is created to verify that non compressed payloads (in case it happens) are also processed and delivered to the consumer.*

**Normative source**:

[e-SENS]

"*Due to the mandatory use of AS4 compression (see paragraph below), XML Payloads are exchanged as compressed binary data*"

**Target**:

Payload compression mandatory

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile. (One-Way/Push MEP)
- SMSH is simulated to produce uncompressed payloads.
- The SMSH sends the AS4 message to the RMSH.

**Predicate**:

The RMSH returns a non-repudiation receipt and delivers the message to the consumer.

**Prescription Level**:

Mandatory

**Tag**:

Payload, compression

**Variable**:

**TA id**:

eSENS_TA05

**Normative source**:

[e-SENS]

"*Due to the mandatory use of AS4 compression (see paragraph below), XML Payloads are exchanged as compressed binary data, which is carried in separate MIME parts and not in the SOAP Body. Therefore, AS4 messages based on this profile always have an empty SOAP Body.*"

**Target**:

Payload location

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile. (One-Way/Push MEP)
- Producer submits a message with metadata information and an XML payload to the SMSH.
- SMSH generates an AS4 message to send to the RMSH.

**Predicate**:

In the AS4 message created by the SMSH, the compressed payload is carried in a separate MIME part and the soap body is empty.

**Prescription Level**:

Mandatory

**Tag**:

Payload packaging

**Variable**:

**TA id**:

eSENS_TA06

**Normative source**:

[e-SENS]

"*Due to the mandatory use of AS4 compression (see paragraph below), XML Payloads are exchanged as compressed binary data, which is carried in separate MIME parts and not in the SOAP Body. Therefore, AS4 messages based on this profile always have an empty SOAP Body.*"

And

"*A single AS4 UserMessage must reference, via the PayloadInfo header, a single structured business document and may reference one or more other (structured or unstructured) payload parts. The business document is considered the "leading" payload part for business processing.*"

**Target**:

Payload location

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile. (One-Way/Push MEP)
- Producer submits a message to the SMSH with metadata information, an XML payload (leading business document) and other payloads (XML and non XML).
- SMSH generates an AS4 message to send to the RMSH.

**Predicate**:

In the AS4 message created by the SMSH, the compressed payloads are carried in separate MIME parts and the soap body is empty.

**Prescription Level**:

Mandatory

**Tag**:

Payload packaging

**Variable**:

**TA id**:

eSENS_TA07

**Normative source**:

[e-SENS]

"*A single AS4 UserMessage must reference, via the PayloadInfo header, a single structured business document and may reference one or more other (structured or unstructured) payload parts. The business document is considered the "leading" payload part for business processing. Any payload parts other than the business document are not to be processed in isolation but only as adjuncts to the business document.*"

**Target**:

Payload processing

**Prerequisite**:

- eSENS_TA06
- SMSH sends an AS4 message to the RMSH.

**Predicate**:

The RMSH successfully processes the AS4 message and sends a non-repudiation receipt to the SMSH.

**Prescription Level**:

Mandatory

**Tag**:

Payload packaging

**Variable**:

**TA id**:

eSENS_TA08

*Note:*

*Not testable.*

**Normative source**:

[e-SENS]

"*A single AS4 UserMessage must reference, via the PayloadInfo header, a single structured business document and may reference one or more other (structured or unstructured) payload parts. The business document is considered the "leading" payload part for business processing. Any payload parts other than the business document are not to be processed in isolation but only as adjuncts to the business document.*"

**Target**:

Payload processing

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile. (One-Way/Push MEP)
- SMSH is simulated to send an AS4 message to the RMSH with non XML payloads and without a leading business document payload.
- The SMSH sends the AS4 User Message to the RMSH.

**Predicate**:

The RMSH sends back a synchronous error response.

**Prescription Level**:

Mandatory

**Tag**:

Payload packaging

**Variable**:

**TA id**:

eSENS_TA09

**Normative source**:

[e-SENS]

"*The ebMS3 mechanism of supporting "external" payloads via hyperlink references (as mentioned in section 5.2.2.12 of the ebMS3 Core Specification [EBMS3]) must not be used.*"

**Target**:

Payload location

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH is simulated to send an AS4 user message with a payload hyperlink reference.

**Predicate**:

The RMSH sends back a synchronous ebMS error message.

**Prescription Level**:

Mandatory

**Tag**:

Payload reference

**Variable**:

**TA id**:

eSENS_TA10

**Normative source**:

[e-SENS]

"*This profile requires the use of the AS4 Reception Awareness feature. This feature provides a built-in Retry mechanism that can help overcome temporary network or other issues and detection of message duplicates.*

•*The parameter PMode[1].ReceptionAwareness must be set to true.*

•*The parameter PMode[1].ReceptionAwareness.Retry must be set to true.*

•*The parameter PMode[1].ReceptionAwareness.DuplicateDetection must be set to true.*

*The parameters PMode[1].ReceptionAwareness.Retry.Parameters and related PMode[1].ReceptionAwareness.DuplicateDetection.Parameters are sets of parameters configuring retries and duplicate detection. These parameters are not fully specified in [AS4] and implementation-dependent. Products must support configuration of parameters for retries and duplicate detection.*"

**Target**:

Message reliability

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Simulate the RMSH to not send receipts (can be done by intercepting the receipts).
- SMSH tries to send an AS4 User Message to the RMSH.

**Predicate**:

The SMSH retries to send the AS4 User Message (at least once).

**Prescription Level**:

Mandatory

**Tag**:

Reception Awareness

**Variable**:

**TA id**:

eSENS_TA11

**Normative source**:

[e-SENS]

"*This profile requires the use of the AS4 Reception Awareness feature. This feature provides a built-in Retry mechanism that can help overcome temporary network or other issues and detection of message duplicates.*

*The parameters PMode[1].ReceptionAwareness.Retry.Parameters and related PMode[1].ReceptionAwareness.DuplicateDetection.Parameters are sets of parameters configuring retries and duplicate detection. These parameters are not fully specified in [AS4] and implementation-dependent. Products must support configuration of parameters for retries and duplicate detection.* "

**Target**:

Message reliability

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Simulate the RMSH to not send receipts.
- SMSH tries to send an AS4 User Message to the RMSH.
- Before a TIME_OUT is reached network connection is restored (RMSH is able to send a non-repudiation receipt).

**Predicate**:

The RMSH sends back an AS4 non-repudiation receipt to the SMSH and delivers only one message to the consumer and the SMSH stops resending the original AS4 User Message.

**Prescription Level**:

Mandatory

**Tag**:

Reception Awareness

**Variable**:

TIME_OUT: deadline (in terms of time or number of retries) allocated for resending messages.

**TA id**:

eSENS_TA12

*Note:*

*Not testable.*

**Normative source**:

[e-SENS]

"*The parameter PMode[1].ErrorHandling.Report.SenderErrorsTo must not be set. There is no support for reporting sender errors to a third party*"

**Target**:

Message reliability

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile.

**Predicate**:

PMode parameter " PMode[1].ErrorHandling.Report.SenderErrors" is not set.

**Prescription Level**:

Mandatory

**Tag**:

Error report

**Variable**:

**TA id**:

eSENS_TA13

**Normative source**:

[e-SENS]

"*This e-SENS AS4 profile uses the following AS4 parameters and values:*

•*The PMode[1].Security.X509.Sign parameter must be set in accordance with section 5.1.4 and 5.1.5 of [AS4].*

•*The PMode[1].Security.X509.Signature.HashFunction parameter must be set to http://www.w3.org/2001/04/xmlenc#sha256.*

•*The PMode[1].Security.X509.Signature.Algorithm parameter must be set to http://www.w3.org/2001/04/xmldsig-more#rsa-sha256.*"

And

"The PMode[1].Security.X509.Signature.Certificate parameter must be set to a value matching the certificate of the sender.

The PMode[1].Security.X509.Encryption.Certificate parameter must be set to a value matching the certificate of the receiver."

**Target**:

Payload signature

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a message with metadata information and a payload to the SMSH.
- SMSH sends an AS4 signed message to the RMSH.

**Predicate**:

In the AS4 Message generated by the SMSH:

- Signature Hash function parameter is set to http://www.w3.org/2001/04/xmlenc#sha256

- Signature Algorithm parameter is set to http://www.w3.org/2001/04/xmldsig-more#rsa-sha256

- Signature Certificate used is the certificate of the SMSH.

**Prescription Level**:

Mandatory

**Tag**:

Signature/Compression

**Variable**:

**TA id**:

eSENS_TA14

**Normative source**:

[e-SENS]

"*This e-SENS AS4 profile uses the following AS4 parameters and values:*

*For encryption, WS-Security leverages the W3C XML Encryption recommendation. The following AS4 configuration options configure this feature:*

•*The PMode[1].Security. X509.Encryption.Encrypt parameter must be set in accordance with section 5.1.6 and 5.1.7 of [AS4].*

•*The parameter PMode[1].Security.X509.Encryption.Algorithm must be set to http://www.w3.org/2009/xmlenc11#aes128-gcm. This is the algorithm used as value for the Algorithm attribute of xenc:EncryptionMethod on xenc:EncryptedData.*"

And

*"The PMode[1].Security.X509.Encryption.Certificate parameter must be set to a value matching the certificate of the receiver."*

**Target**:

Payload signature

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a message with metadata information and a payload to the SMSH.
- SMSH sends an AS4 encrypted message to the RMSH.

**Predicate**:

In the AS4 Message generated by the SMSH:

- Encryption Algorithm is set to http://www.w3.org/2009/xmlenc11#aes128-gcm

- Encryption Certificate used is the certificate of the RMSH.

**Prescription Level**:

Mandatory

**Tag**:

Signature/Compression

**Variable**:

**TA id**:

eSENS_TA15

**Normative source**:

[e-SENS]

"*To be able to forward a received message, the receiving gateway (C3) needs to be able to determine the end entity (C4) that an e-SENS AS4 message is intended for. This information is generally available in the business document. However, using information from the business document assumes an understanding of the schema on which the document is based. Since gateways need to be able to process documents of any type, it is desirable to adopt a mechanism that is independent of particular schemas.*

*The e-CODEX documentation for its use of ebMS3/AS4 [ECODEXD59] uses the ebMS3 property mechanism to attach arbitrary pairs of property-values to a message to address C1 and C4:*

*•The property named originalSender addresses the original (end entity) sender party.*

*•The property named finalRecipient addresses the final (end entity) recipient party*

*The type attribute may be used to categorize party identifier types. Implementations of the e-SENS e-Delivery AS4 profile must support this mechanism: the sender gateway (or integration middleware) must set, and the receiver gateway (or integration middleware) must get, these properties and values.*"

**Target**:

Messaging Reliability

**Prerequisite**:

- Producer submits a business document with the information “Sender” and “destination” to the SMSH
- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH sends an AS4 User message to the RMSH.

**Predicate**:

The message received by the RMSH contains 2 property elements in the MessageProperties node with attributes name and type. One has name = "OriginalSender" and value producerID and the other has name = "finalRecipient" and value consumerID (producerID and consumerID are provided by the original message submitted by the producer).

**Prescription Level**:

Mandatory

**Tag**:

User message, end entity Addressing

**Variable**:

producerID: ID of the producer of the AS4 message

consumerID: ID of the consumer of the message

**TA id**:

eSENS_TA16

**Normative source**:

[eSENS]

"*For Reliable Messaging this profile specifies that non-repudiation receipts must be sent synchronously for each message type. Note that non-repudiation is only "per hop" in the case of the four-corner-model, in particular the hop from corner two to corner three. In e-SENS, the optional end-to-end services module supports the traceability across the four corners.*

•*The parameter PMode[1].Security.SendReceipt.NonRepudiation must be set to the value true.*

•*The parameter PMode[1].Security.SendReceipt.ReplyPattern must be set to the value Response.*"

**Target**:

Message reliability

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (Two-Way/Push-and-Push MEP).
- SMSH sends an AS4 User Message to the RMSH.

**Predicate**:

The RMSH returns a non-repudiation receipt within a HTTP response with status code 2XX.

**Prescription Level**:

Mandatory

**Tag**:

Non-repudiation receipt

**Variable**:

**TA id**:

eSENS_TA17

**Normative source**:

[eSENS]

"*Reception awareness errors generated by the Sender must be reported to the Submitting application:*

•*The parameter PMode[1].ErrorHandling.Report.MissingReceiptNotifyProducer must be set to true.*

•*The parameter PMode[1].ErrorHandling.Report.SenderErrorsTo must not be set. There is no support for reporting sender errors to a third party*"

**Target**:

Message reliability

**Prerequisite**:

- eSENS_TA10.
- TIME_OUT for resending the messages is reached.

**Predicate**:

The SMSH reports an error to the message producer.

**Prescription Level**:

Mandatory

**Tag**:

Reception Awareness

**Variable**:

TIME_OUT: deadline (in terms of time or number of retries) allocated for resending messages.

**TA id**:

eSENS_TA18

**Normative source**:

[eSENS]

" *Section 5.2.2 of [EBMS3] defines a server test feature that allows an organization to "Ping" a communication partner. The feature is based on messages with the values of:*

*UserMessage/CollaborationInfo/Service set to http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service*

*UserMessage/CollaborationInfo/Action set to http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test*

*This feature must be supported so that business partners can perform a basic test of the communication configuration (including security at network, transport and message layer, and reliability) in any environment, including the production environment. This functionality may be supported as a built-in feature of the AS4 product. If not, a PMode must be configured with these values.* "

**Target**:

Test service

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- Producer submits a "ping" message with metadata information to the SMSH (to "ping" consumer).

**Predicate**:

The SMSH generates an AS4 message with values (and sends it to the RMSH):

- **UserMessage/CollaborationInfo/Service** set to http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/service
- **UserMessage/CollaborationInfo/Action** set to http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/test

**Prescription Level**:

Mandatory

**Tag**:

Ping message

**Variable**:

**TA id**:

eSENS_TA19

**Normative source**:

[eSENS]

"*The AS4 product must be configured so that messages with these values are not delivered to any business application.*"

**Target**:

Test service

**Prerequisite**:

- eSENS_TA18.
- The consumer is reachable.

**Predicate**:

The RMSH sends back a non-repudiation receipt within a HTTP response with status code 2XX and the consumer doesn't receive any message.

**Prescription Level**:

Mandatory

**Tag**:

Ping message

**Variable**:

**TA id**:

eSENS_TA20

*Note:*

*This is a new test assertion: not existing in the Draft5.*

**Normative source**:

[eSENS]

*"This profile defines an additional, optional third property:*

- *The property named trackingIdentifier provides a mechanism to include an identifier (in arbitrary string format) that allows end-to-end tracking of messages in a four-corner exchange. Its value could be set to the value of an identifier for the message from C1 to C2 that the AS4 message relates to. This allows tracking and tracing of messages from C1, via C2 to (at least) C3."*

**Target**:

Message Exchange tracking

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- SMSH sends an AS4 User Message including a TRACKINGIDENTIFIER property set by the producer.

**Predicate**:

The RMSH returns a non-repudiation receipt within a HTTP response with status code 2XX and the received AS4 message contains the TRACKINGIDENTIFIER property.

**Prescription Level**:

Optional

**Tag**:

Message Packaging, Tracking, TrackingIdentifier

**Variable**:

TRACKINGIDENTIFIER: an arbitrary string format.

**Generic AS4 test assertions**

**TA id**:

AS4_TA01

**Normative source**:

[eSENS]

"*AS4 provides multiple mechanisms to correlate messages within a particular flow.*

*1.UserMessage/MessageInfo/RefToMessageId provides a way to express that a message is a response to a single specific previous message. Presence of a RefToMessageId is required in response messages in Two Way message exchanges. By default, exchanges are considered One Way.*

*2.UserMessage/CollaborationInfo/ConversationId provides a more general way to associate a message with an ongoing conversation, without requiring a message to be a response to a single specific previous message, but allowing update messages to existing conversations from both Sender and Receiver of the original message.*"

**Target**:

Message Exchange conversation

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: Two-Way/Push-and-Push MEP.
- SMSH sends an AS4 User Message (M1) associated to a specific conversation through variable (element) CONVERSATIONIDM1 (set by the producer).
- The consumer replies to the message M1.

**Predicate**:

The RMSH sends back a User Message (M2) with element CONVERSATIONIDM2 equal to ConversationIdM1 (set by the consumer).

**Prescription Level**:

Mandatory

**Tag**:

Message Packaging, Correlation, Conversation ID

**Variable**:

CONVERSATIONIDM1: XML element (M1) UserMessage/CollaborationInfo/ConversationId

CONVERSATIONIDM2: XML element (M2) UserMessage/CollaborationInfo/ConversationId

**TA id**:

AS4_TA02

**Normative source**:

[eSENS]

"*A compliant product must allow the Producer, when submitting messages, to set values for MessageId, RefToMessageId and ConversationId, to support correlation).*"

**Target**:

User Message exchange parameters

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- Producer submits a Message including metadata information and payload to the SMSH with setting message parameters: MESSAGEID, REFTOMESSAGEID and CONVERSATIONID.

**Predicate**:

The SMSH returns a successful submission Notification and the AS4 Message generated by the SMSH contains the same parameters values set by the producer.

**Prescription Level**:

Mandatory

**Tag**:

User Message

**Variable**:

MESSAGEID: XML element Messaging/UserMessage/MessageInfo/MessageId

REFTOMESSAGEID: XML element Messaging/UserMessage/MessageInfo/ RefToMessageId

CONVERSATIONID: XML element Messaging/UserMessage /CollaborationInfo/ConversationId

**TA id**:

AS4_TA03

*Note:*

*Not yet implemented (https://issues.oasis-open.org/browse/EBXMLMSG-2). The prescription level is downgraded to optional till the implementation.*

**Normative source**:

[eSENS]

"*As in the AS4 ebHandler profile, support for MessageProperties is required in this profile. It must be possible to set the type attribute for message properties*"

and

"https://issues.oasis-open.org/browse/EBXMLMSG-2"

**Target**:

User Message properties

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH is simulated to send an AS4 User message to the RMSH with parameter MESSAGEPROPERTIES containing:
    - A property with attributes "name" and "type" present.
    - A property with only attribute "name" present.

**Predicate**:

The RMSH returns a non-repudiation receipt within a HTTP response with status code 2XX.

**Prescription Level**:

Optional

**Tag**:

User Message

**Variable**:

MESSAGEPROPERTIES: XML element Messaging/UserMessage/MessageProperties with children nodes format <Property name="" type=""> </Property>

**TA id**:

AS4_TA04

**Normative source**:

[eSENS]

"*Section 5.1.1 of the ebMS3 Core Specification [EBMS3] requires implementations to process both non-multipart (simple SOAP) messages and multipart (SOAP-with-attachments) messages*"

*and this is a requirement for the AS4 ebHandler Conformance Profile.*

**Target**:

AS4 Message format

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile: One-Way/Push MEP.
- SMSH sends an AS4 message (User Message with payload) to the RMSH.

**Predicate**:

The RMSH sends a non-repudiation receipt to the SMSH.

**Prescription Level**:

Mandatory

**Tag**:

Message format, Message packaging, SOAP-with-attachments

**Variable**:

**TA id**:

AS4_TA05

**Normative source**:

[eSENS]

"*Section 5.1.1 of the ebMS3 Core Specification [EBMS3] requires implementations to process both non-multipart (simple SOAP) messages and multipart (SOAP-with-attachments) messages*"

**Target**:

AS4 Message format

**Prerequisite**:

- AS4_TA04

**Predicate**:

The SMSH sends a success notification to the producer.

**Prescription Level**:

Mandatory

**Tag**:

Message format, simple SOAP

**Variable**:

**TA id**:

AS4_TA06

**Normative source**:

[eSENS]

"*Due to the mandatory use of AS4 compression (see paragraph below), XML Payloads are exchanged as compressed binary data*"

**Target**:

Payload compression mandatory

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a Message with metadata information and XML payload to the SMSH.

**Predicate**:

The SMSH generates an AS4 message with a gzip compressed payload.

**Prescription Level**:

Mandatory

**Tag**:

Payload, compression

**Variable**:

**TA id**:

AS4_TA07

**Normative source**:

[eSENS]

"*The PartInfo element in the message header that relates to the compressed message part, must have an Property element with @name ="CompressionType" … The content type of the compressed attachment must be "application/gzip*""

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a Message with metadata information and payload to the SMSH.

**Predicate**:

In the AS4 message generated by the SMSH, a property element with name "CompressionType" and value set to "application/gzip" is present.

**Prescription Level**:

Mandatory

**Tag**:

Payload, compression, compression Type

**Variable**:

**TA id**:

AS4_TA08

**Normative source**:

[eSENS]

"*Packaging requirements:*

*•An PartInfo/PartProperties/Property/@name="MimeType" value is required to identify the MIME type of the payload before compression was applied.*"

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a message to the SMSH with payload (ex: xml document) and metadata information including a property element with name "MimeType" and value ("application/xml").

**Predicate**:

The SMSH generates an AS4 message with the property "MimeType" present and set to the value specified by the producer ("application/xml").

**Prescription Level**:

Mandatory

**Tag**:

Payload, compression, Mime Type

**Variable**:

**TA id**:

AS4_TA09

**Normative source**:

[eSENS]

"*Packaging requirements:*

*•An PartInfo/PartProperties/Property/@name="MimeType" value is required to identify the MIME type of the payload before compression was applied.*"

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- The SMSH is simulated to send an AS4 message without property "MimeType" present to the RMSH.

**Predicate**:

The RMSH sends a synchronous ebMS error response.

**Prescription Level**:

Mandatory

**Tag**:

Payload, compression, Mime Type

**Variable**:

**TA id**:

AS4_TA10

**Normative source**:

[eSENS]

"*Packaging requirements:*

*•For XML payloads, an PartInfo/PartProperties/Property/@name="CharacterSet" value is recommended to identify the character set of the payload before compression was applied. The value of this property MUST conform to the values defined in section 4.3.3 of [XML10].*"

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a message to the SMSH with xml (UTF-16) payload and metadata information including payload characterset info.

**Predicate**:

The SMSH generates an AS4 message with the property "CharacterSet" present and set to the value "UTF-16".

**Prescription Level**:

Preferred

**Tag**:

Payload, compression, Character Set

**Variable**:

**TA id**:

AS4_TA11

**Normative source**:

[eSENS]

"*Packaging requirements:*

*•For XML payloads, an PartInfo/PartProperties/Property/@name="CharacterSet" value is recommended to identify the character set of the payload before compression was applied. The value of this property MUST conform to the values defined in section 4.3.3 of [XML10].*"

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits a message to the SMSH with xml (UTF-8) payload and metadata information including payload characterset info.

**Predicate**:

The SMSH generates an AS4 message with the property "CharacterSet" present and set to the value "UTF-8".

**Prescription Level**:

Preferred

**Tag**:

Payload, compression, Character Set

**Variable**:

**TA id**:

AS4_TA12

*Note:*

*Not testable (might become valid after the requirements EBXMLMSG-87 and EBXMLMSG-88 are validated).*

**Normative source**:

[eSENS]

"*Packaging requirements:*

*•For XML payloads, an PartInfo/PartProperties/Property/@name="CharacterSet" value is recommended to identify the character set of the payload before compression was applied. The value of this property MUST conform to the values defined in section 4.3.3 of [XML10].*"

*And*

"*https://issues.oasis-open.org/browse/EBXMLMSG-87*"

*And*

"*https://issues.oasis-open.org/browse/EBXMLMSG-88*"

**Target**:

Payload compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH is simulated to send an AS4 message with property element "CharacterSet" set to value not conform to section 4.3.3 of [XML10] (example: "!utf*").
- The SMSH sends the AS4 message to the RMSH.

**Predicate**:

The RMSH returns a synchronous ebMS error message.

**Prescription Level**:

mandatory

**Tag**:

Payload, compression, Character Set

**Variable**:

**TA id**:

AS4_TA13

**Normative source**:

[eSENS]

"*In case of error during decompression, the following error MUST be used: Code = EBMS:0303, Short Description = DecompressionFailure, Severity = Failure, Category = Communication. "*

*"Error Handling*

*For the error handling this profile specifies that errors must be reported and transmitted synchronously to the Sender and should be reported to the Consumer.*

*•The parameter PMode[1].ErrorHandling.Report.AsResponse must be set to the value true."*

**Target**:

Message compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH is simulated to send an AS4 User Message with compressed but damaged payloads.
- The SMSH sends the AS4 User Message to the RMSH.

**Predicate**:

The RMSH sends back a synchronous error response with error code "Code = EBMS:0303, Short Description = DecompressionFailure, Severity = Failure, Category = Communication".

**Prescription Level**:

Mandatory

**Tag**:

Payload Compression, error handling

**Variable**:

**TA id**:

AS4_TA14

**Normative source**:

[eSENS]

"*In case of error during decompression, the following error MUST be used: Code = EBMS:0303, Short Description = DecompressionFailure, Severity = Failure, Category = Communication. "*

*"Error Handling*

*For the error handling this profile specifies that errors must be reported and transmitted synchronously to the Sender and should be reported to the Consumer.*

*•The parameter PMode[1].ErrorHandling.Report.AsResponse must be set to the value true.*

*•The parameter PMode[1].ErrorHandling.Report.ProcessErrorNotifyConsumer should be set to the value true.* "

**Target**:

Message compression

**Prerequisite**:

- AS4_TA13.
- The User Message is bound to a PMode with parameter PMode[1].ErrorHandling.Report.ProcessErrorNotifyConsumer: set to true.

**Predicate**:

The RMSH sends back a synchronous error response (with error code "Code = EBMS:0303, Short Description = DecompressionFailure, Severity = Failure, Category = Communication") and the RMSH sends an error notification to the consumer of the message.

**Prescription Level**:

Preferred

**Tag**:

Payload Compression error

**Variable**:

**TA id**:

AS4_TA15

**Normative source**:

[eSENS]

"*The receiving AS4 MSH must decompress any payload part(s) compressed by the SMSH before delivering the message.*"

**Target**:

Payload reception

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH sends an AS4 User Message with a compressed payload to the RMSH.

**Predicate**:

The RMSH delivers the message with decompressed payload to the consumer.

**Prescription Level**:

Mandatory

**Tag**:

Payload delivery

**Variable**:

**TA id**:

AS4_TA16

**Normative source**:

[eSENS]

"*The receiving AS4 MSH must decompress any payload part(s) compressed by the SMSH before delivering the message.*"

**Target**:

Payload reception

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH sends an AS4 User Message with a several compressed payloads (XML and non XML) to the RMSH.

**Predicate**:

The RMSH delivers the message with decompressed payloads to the consumer.

**Prescription Level**:

Mandatory

**Tag**:

Payload delivery

**Variable**:

**TA id**:

AS4_TA17

**Normative source**:

[eSENS]

"*When compression, signature and/or encryption are required, any attached payload(s) must be compressed prior to being signed and/or encrypted.*"

**Target**:

Payload compression

**Prerequisite**:

- eSENS_TA13.
- The SMSH is simulated to send an AS4 User message with a compressed then signed payload to the RMSH.

**Predicate**:

The RMSH sends back an AS4 non-repudiation receipt.

**Prescription Level**:

Mandatory

**Tag**:

Signature/Compression

**Variable**:

**TA id**:

AS4_TA18

**Normative source**:

[eSENS]

"*When compression, signature and/or encryption are required, any attached payload(s) must be compressed prior to being signed and/or encrypted.*"

**Target**:

Message level security

**Prerequisite**:

- eSENS_TA13
- Simulated SMSH sends a signed AS4 User Message with a signed then compressed payload to the RMSH.

**Predicate**:

The SMSH receives a WS-Security SOAP Fault.

**Prescription Level**:

Mandatory

**Tag**:

Encryption/Signing Compression

**Variable**:

**TA id**:

AS4_TA19

**Normative source**:

[eSENS]

"*When compression, signature and/or encryption are required, any attached payload(s) must be compressed prior to being signed and/or encrypted.*"

**Target**:

Payload encryption

**Prerequisite**:

- eSENS_TA14.
- The SMSH is simulated to send a compressed then encrypted AS4 message to the RMSH.

**Predicate**:

The RMSH sends back an AS4 non-repudiation receipt.

**Prescription Level**:

Mandatory

**Tag**:

encryption/Compression

**Variable**:

**TA id**:

AS4_TA20

**Normative source**:

[eSENS]

"*When compression, signature and/or encryption are required, any attached payload(s) must be compressed prior to being signed and/or encrypted.*"

**Target**:

Message level security

**Prerequisite**:

- eSENS_TA14
- Simulated SMSH sends a signed AS4 User Message with an encrypted first, then compressed payload to the RMSH.

**Predicate**:

The SMSH receives a WS-Security SOAP Fault.

**Prescription Level**:

Mandatory

**Tag**:

Encryption/Signing Compression

**Variable**:

**TA id**:

AS4_TA21

**Normative source**:

[eSENS]

"*When compression, signature and/or encryption are required, any attached payload(s) must be compressed prior to being signed and/or encrypted.*"

**Target**:

Payload signature/encryption/compression

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- The SMSH sends an AS4 message with a compressed then encrypted and signed payload to the RMSH.

**Predicate**:

The RMSH sends back an AS4 non-repudiation receipt.

**Prescription Level**:

Mandatory

**Tag**:

Signature/encryption/compression

**Variable**:

**TA id**:

AS4_TA22

*Note:*

*This test assertion is only valid in case TLS is handled by the AS4 message handler.*

**Normative source**:

[eSENS]

*"•It must be possible to configure the accepted TLS version(s) in the AS4 message handler.*

*•It must be possible to configure accepted TLS cipher suites in the AS4 message handler.*"

**Target**:

Transport Layer security

**Prerequisite**:

**Predicate**:

Parameters to configure TLS version and cipher suites exist.

**Prescription Level**:

Mandatory

**Tag**:

TLS

**Variable**:

**TA id**:

AS4_TA23

*Note:*

*This test assertion is only valid in case TLS is handled by the AS4 message handler.*

**Normative source**:

[eSENS]

"*The ENISA and BSI reports state that TLS 1.0 and TLS 1.1 should not be used in new applications. Older version such as SSL 2.0 [RFC6176] and SSL 3.0 must not be used. Products compliant with this profile should therefore support TLS 1.2 [RFC5246].*"

**Target**:

Transport Layer security

**Prerequisite**:

- AS4_TA23
- RMSH is configured with TLS version 1.2
- SMSH is configured with TLS1.0 or TLS1.1 version
- SMSH tries to submit an AS4 message to the RMSH.

**Predicate**:

Connection is not established between the SMSH and the RMSH..

**Prescription Level**:

Preferred

**Tag**:

TLS

**Variable**:

**TA id**:

AS4_TA24

*Note:*

*This test assertion is only valid in case TLS is handled by the AS4 message handler.*

**Normative source**:

[eSENS]

"*The ENISA and BSI reports state that TLS 1.0 and TLS 1.1 should not be used in new applications. Older version such as SSL 2.0 [RFC6176] and SSL 3.0 must not be used. Products compliant with this profile should therefore support TLS 1.2 [RFC5246].*"

**Target**:

Transport Layer security

**Prerequisite**:

- AS4_TA22
- RMSH is configured with TLS version 1.2
- SMSH is configured with SSL2.0 or SSL3.0 version
- SMSH tries to submit an AS4 message to the RMSH.

**Predicate**:

Connection is not established between the SMSH and the RMSH.

**Prescription Level**:

Mandatory

**Tag**:

TLS, Error

**Variable**:

**TA id**:

AS4_TA25

*Note:*

*This test assertion is only valid in case TLS is handled by the AS4 message handler.*

**Normative source**:

[eSENS]

*"•IANA publishes a list of TLS cipher suites [TLSSP], only a subset of which the ENISA Report considers future-proof (see [ENISAAKSP], section 5.1.2). Products must support cipher suites included in this subset. Vendors must add support for newer, safer cipher suites, as and when such suites are published by IANA/IETF.*

*•Support for SSL 3.0 and for cipher suites that are not currently considered secure should be disabled by default.*

*•Perfect Forward Secrecy, which is required in [BSITLS], is supported by the TLS_ECDHE_* and TLS_DHE_* cipher suites, which are therefore preferred and should be supported.*"

**Target**:

Transport Layer security

**Prerequisite**:

- AS4_TA22
- RMSH is configured with TLS version 1.2 and list_accepted_cipher_ suites.
- SMSH is configured with TLS 1.2 and cipher_suites not in list_accepted_cipher_ suites.
- SMSH submits an AS4 message to the RMSH.

**Predicate**:

Connection is not established between the SMSH and the RMSH.

**Prescription Level**:

Mandatory

**Tag**:

TLS

**Variable**:

list_accepted_cipher_ suites: subset of list of TLS cipher suites [TLSSP] and TLS_ECDHE_* and TLS_DHE_* cipher suites

**TA id**:

AS4_TA26

**Normative source**:

[eSENS]

*"•IANA publishes a list of TLS cipher suites [TLSSP], only a subset of which the ENISA Report considers future-proof (see [ENISAAKSP], section 5.1.2). Products must support cipher suites included in this subset. Vendors must add support for newer, safer cipher suites, as and when such suites are published by IANA/IETF.*

*•Support for SSL 3.0 and for cipher suites that are not currently considered secure should be disabled by default.*

*•Perfect Forward Secrecy, which is required in [BSITLS], is supported by the TLS_ECDHE_* and TLS_DHE_* cipher suites, which are therefore preferred and should be supported.*"

**Target**:

Transport Layer security

**Prerequisite**:

- AS4_TA22
- RMSH is configured with TLS version 1.2 and list_accepted_cipher_ suites.
- SMSH is configured with TLS version 1.2 and cipher_suites in list_accepted_cipher_ suites.
- SMSH submits an AS4 message to the RMSH.

**Predicate**:

The RMSH returns an HTTP response code 2XX.

**Prescription Level**:

Mandatory

**Tag**:

TLS

**Variable**:

list_accepted_cipher_ suites: subset of list of TLS cipher suites [TLSSP] and TLS_ECDHE_* and TLS_DHE_* cipher suites

**TA id**:

AS4_TA27

**Normative source**:

[eSENS]

"*From/PartyId and To/PartyId shall therefore in this case address the identifiers of gateways. This is consistent with current practice for ebMS3 in e-CODEX and with the PEPPOL AS2 profile*"

**Target**:

Messaging Reliability

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- SMSH sends an AS4 User message to the RMSH.

**Predicate**:

In the message sender|receiver elements reference the MSHs and not the (producer,consumer).

**Prescription Level**:

Mandatory

**Tag**:

Adressing

**Variable**:

sender|receiver: UserMessage/PartyInfo/{From|to}/PartyId elements

**TA id**:

AS4_TA28

*Note:*

*This test assertion is valid (and then mandatory) only in case SBDH is used.*

**Normative source**:

[eSENS]

"*An option for e-SENS is to use UN/CEFACT Standard Business Document Header [SBDH] that allows end entities to encode information on business process, business transaction, agreement, and business quality-of-service. The SBDH is widely adopted in e-business communities like GS1 […] for e-SENS should be used a Manifest block for sending non-XML documents or files. […]When sending non-XML documents the SBDH and the payload have to be in separate MIME parts(see the figure above) because they have different content types: SBDH is XML and the payload non-XML.*"

**Target**:

SBDH

**Prerequisite**:

- SMSH and RMSH are configured to exchange AS4 messages according to the eSENS profile (One-Way/Push MEP).
- Producer submits two payloads, first being an SBDH document, second being an actual payload (non-XML payload).
- SMSH sends an AS4 User Message to the RMSH.

**Predicate**:

Message has two additional MIME parts. The first mime part is the SBDH document and the second is the actual payload.

**Prescription Level**:

Mandatory

**Tag**:

Message Packaging

**Variable**:

SDBH: information on end point IDs, business process, business transaction, agreement and business quality.