International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.830.1

(04/2017)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

E-health multimedia services and applications – Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)

Conformance of ITU-T H.810 personal health system: Services interface Part 1: Web services interoperability: Health & Fitness Service sender

Recommendation ITU-T H.830.1

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	H.260–H.279
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence	H.420–H.429
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
Ubiquitous sensor network applications and Internet of Things	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	H.700–H.719
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	**H.820–H.859**
Multimedia e-health data exchange services	H.860–H.869

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T H.830.1

Conformance of ITU-T H.810 personal health system: Services interface Part 1: Web services interoperability: Health & Fitness Service sender

Summary

Recommendation ITU-T H.830.1 provides a test suite structure (TSS) and the test purposes (TP) for Web services interoperability for messages through the Health & Fitness Service (HFS) sender in the Services interface, based on the requirements defined in the Recommendations of the ITU-T H.810 sub-series, of which Recommendation ITU-T H.810 (2016) is the base Recommendation. The objective of this test specification is to provide a high probability of interoperability at this interface.

Recommendation ITU-T H.830.1 is a transposition of Continua Test Tool DG2016, Test Suite Structure & Test Purposes, Services Interface; Part 1: Web Services Interoperability. HFS Sender (Version 1.6, 2017-03-14), that was developed by the Personal Connected Health Alliance. A number of versions of this specification existed before transposition.

This Recommendation includes an electronic attachment with the protocol implementation conformance statements (PICS) and the protocol implementation extra information for testing (PIXIT) required for the implementation of Annex A.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T H.831	2015-01-13	16	11.1002/1000/12249
1.0	ITU-T H.830.1	2015-01-13	16	11.1002/1000/12587
2.0	ITU-T H.830.1	2016-07-14	16	11.1002/1000/12921
3.0	ITU-T H.830.1	2017-04-13	16	11.1002/1000/13201

Keywords

Conformance testing, Continua Design Guidelines, e-health, Health & Fitness Service sender, ITU-T H.810, personal connected health devices, Services interface, Web services interoperability.

---

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Page

Electronic attachment: This Recommendation includes an electronic attachment with the protocol implementation conformance statements (PICS) and the protocol implementation extra information for testing (PIXIT) required for the implementation of Annex A.

Introduction

This Recommendation is a transposition of Continua Test Tool DG2016, Test Suite Structure & Test Purposes, Services Interface; Part 1: Web Services Interoperability. HFS Sender (Version 1.6, 2017-03-14), that was developed by the Personal Connected Health Alliance. The table below shows the revision history of this test specification; it may contain versions that existed before transposition.

Version	Date	Revision history
1.2	2012-10-05	Initial release for Test Tool DG2011. It is the same version as "TSS&TP_1.5_WAN_PART_1_(SEN WS-I)_v1.2.doc" because new features included in [b-CDG 2011] do not affect the test procedures specified in this document.
1.2	2013-05-24	Initial release for Test Tool DG2012. It is the same version as "TSS&TP_DG2011_WAN_PART_1_(SEN WS-I)_v1.2.doc" because new features included in [b-CDG 2012] do not affect the test procedures specified in this document.
1.2	2014-01-24	Initial release for Test Tool DG2013. It is the same version as "TSS&TP_DG2012_WAN_PART_1_(SEN WS-I)_v1.2.doc" because new features included in CDG 2013 [b-ITU-T H.810 (2013)]/[b-CDG 2013] do not affect the test procedures specified in this document.
1.3	2014-04-24	TM Lite & Doc Enhancements (Test Tool v4.0 Maintenance Release 1). It uses "TSS&TP_DG2013_ WAN_PART_1_(SEN WS-I)_v1.2.doc" as baseline and it adds new features included in Documentation Enhancements: • "Other PICS" row added
1.4	2015-07-01	Initial release for Test Tool DG2015: • Test suite structure modified. • Applicability modified due to the inclusion of hData OU.
1.5	2016-09-20	Initial release for Test Tool DG2016. It implements changes according to [ITU-T H.810 (2016)]/[b-CDG 2016] (Iris + Errata) refreshments.
1.6	2017-03-14	Editorial: added insulin pump and continuous glucose monitor specializations to the TSS list in clause 6.

Recommendation ITU-T H.830.1

Conformance of ITU-T H.810 personal health system: Services interface Part 1: Web services interoperability: Health & Fitness Service sender

1 Scope

The scope of this Recommendation[1] is to provide a test suite structure (TSS) and the test purposes (TP) for the Services interface based on the requirements defined in Continua Design Guidelines (CDG) [ITU-T H.810 (2016)]. The objective of this test specification is to provide a high probability of interoperability at this interface.

The TSS and TP for the Services interface have been divided into the parts specified below. This Recommendation covers Part 1.

– **Part 1: Web services interoperability. Health & Fitness Service sender**
– Part 2: Web services interoperability. Health & Fitness Service receiver
– Part 3: SOAP/ATNA. Health & Fitness Service sender
– Part 4: SOAP/ATNA. Health & Fitness Service receiver
– Part 5: PCD-01 HL7 messages. Health & Fitness Service sender
– Part 6: PCD-01 HL7 messages. Health & Fitness Service receiver
– Part 7: Consent Management. Health & Fitness Service sender
– Part 8: Consent Management. Health & Fitness Service receiver
– Part 9: hData Observation Upload. Health & Fitness Service sender
– Part 10: hData Observation Upload. Health & Fitness Service receiver
– Part 11: Questionnaires. Health & Fitness Service sender
– Part 12: Questionnaires. Health & Fitness Service receiver

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T H.810 (2016)] Recommendation ITU-T H.810 (2016), *Interoperability design guidelines for personal health systems*.

[ITU-T H.812] Recommendation ITU-T H.812 (2016), *Interoperability design guidelines for personal health systems: Services interface: Common certified capability class.*

1 This Recommendation includes an electronic attachment with the protocol implementation conformance statements (PICS) and the protocol implementation extra information for testing (PIXIT) required for the implementation of Annex A.

[ITU-T H.812.1] Recommendation ITU-T H.812.1 (2016), *Interoperability design guidelines for personal health systems: Services interface: Observation upload certified capability class.*

[ITU-T H.812.2] Recommendation ITU-T H.812.2 (2016), *Interoperability design guidelines for personal health systems: Services interface: Questionnaires certified capability class.*

[ITU-T H.812.3] Recommendation ITU-T H.812.3 (2016), *Interoperability design guidelines for personal health systems: Services interface: Capability exchange certified capability class.*

[ITU-T H.812.4] Recommendation ITU-T H.812.4 (2016), *Interoperability design guidelines for personal health systems: Services interface: Authenticated persistent session certified capability class.*

[OASIS/WS-I BP] OASIS/WS-I (2006), *Basic Profile Version 1.1*.
http://www.ws-i.org/Profiles/BasicProfile-1.1.html

[OASIS/WS-I BSP] OASIS/WS-I (2007), *Basic Security Profile Version 1.0*.
http://www.ws-i.org/Profiles/BasicSecurityProfile-1.0.html

[OASIS WS-I RM] OASIS (2007), *Web Services Reliable Messaging (WS-Reliable Messaging) Version 1.1*.
http://docs.oasis-open.org/ws-rx/wsrm/200702/wsrm-1.1-spec-cs-01.pdf

3 Definitions

3.1 Terms defined elsewhere

None.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AHD Application Hosting Device

ATS Abstract Test Suite

ATNA Audit Trail and Node Authentication

CDG Continua Design Guidelines

CGM Continuous Glucose Monitor

DUT Device Under Test

EPR Endpoint Reference

GUI Graphical User Interface

HFS Health & Fitness Service

HFSS Health & Fitness Service Sender

HFSR Health & Fitness Service Receiver

HL7 Health Level 7

HTTP Hypertext Transfer Protocol

HTTPS	Hypertext Transfer Protocol Secure
INR	International Normalized Ratio
IP	Insulin Pump
IUT	Implementation Under Test
MDS	Medical Device System
NFC	Near Field Communication
PCD	Patient Care Device
PCO	Point of Control and Observation
PCT	Protocol Conformance Testing
PHD	Personal Health Device
PHDC	Personal Healthcare Device Class
PHG	Personal Health Gateway
PICS	Protocol Implementation Conformance Statement
PIXIT	Protocol Implementation extra Information for Testing
SABTE	Sleep Apnoea Breathing Therapy Equipment
SCR	Static Conformance Review
SDP	Service Discovery Protocol
SOAP	Simple Object Access Protocol
STR	Security Token Reference
TCRL	Test Case Reference List
TCWG	Test and Certification Working Group
TLS	Transport Level Security
TP	Test Purpose
URI	Uniform Resource Identifier
TSS	Test Suite Structure
USB	Universal Serial Bus
WAN	Wide Area Network
WDM	Windows Driver Model
WS	Web Service
WSDL	Web Service Description Language
XML	extensible Markup Language

5 Conventions

The key words "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "MAY", "MAY NOT" in this Recommendation are to be interpreted as in [b-ETSI SR 001 262].

– SHALL is equivalent to 'must' or 'it is required to'.

– SHALL NOT is equivalent to 'must not' or 'it is not allowed'.

– SHOULD is equivalent to 'it is recommended to'.

– SHOULD NOT is equivalent to 'it is not recommended to'.

– MAY is equivalent to 'is permitted'.

– MAY NOT is equivalent to 'it is not required that'.

NOTE – The above-mentioned key words are capitalized for illustrative purposes only and they do not appear capitalized within this Recommendation.

Reference is made in the ITU-T H.800-series of Recommendations to different versions of the Continua Design Guidelines (CDG) by a specific designation. The list of terms that may be used in this Recommendation is provided in Table 1.

Table 1 – List of designations associated with the various versions of the CDG

CDG release	Transposed as	Version	Description	Designation
2016 plus errata	[ITU-T H.810 (2016)]	6.1	Release 2016 plus errata noting all ratified bugs [b-CDG 2016].	–
2016	–	6.0	Release 2016 of the CDG including maintenance updates of the CDG 2015 and additional guidelines that cover new functionalities.	Iris
2015 plus errata	[b-ITU-T H.810 (2015)]	5.1	Release 2015 plus errata noting all ratified bugs [b-CDG 2015]. The 2013 edition of H.810 is split into eight parts in the H.810-series.	–
2015	–	5.0	Release 2015 of the CDG including maintenance updates of the CDG 2013 and additional guidelines that cover new functionalities.	Genome
2013 plus errata	[b-ITU-T H.810 (2013)]	4.1	Release 2013 plus errata noting all ratified bugs [b-CDG 2013].	–
2013	–	4.0	Release 2013 of the CDG including maintenance updates of the CDG 2012 and additional guidelines that cover new functionalities.	Endorphin
2012 plus errata	–	3.1	Release 2012 plus errata noting all ratified bugs [b-CDG 2012].	–
2012	–	3.0	Release 2012 of the CDG including maintenance updates of the CDG 2011 and additional guidelines that cover new functionalities.	Catalyst
2011 plus errata	–	2.1	CDG 2011 integrated with identified errata.	–
2011	–	2.0	Release 2011 of the CDG including maintenance updates of the CDG 2010 and additional guidelines that cover new functionalities [b-CDG 2011].	Adrenaline
2010 plus errata	–	1.6	CDG 2010 integrated with identified errata	–
2010	–	1.5	Release 2010 of the CDG with maintenance updates of the CDG Version 1 and additional guidelines that cover new functionalities [b-CDG 2010].	1.5

Table 1 – List of designations associated with the various versions of the CDG

CDG release	Transposed as	Version	Description	Designation
1.0	–	1.0	First released version of the CDG [b-CDG 1.0].	–

6 Test suite structure (TSS)

The test purposes (TPs) for the Services interface have been divided into the main subgroups specified below. Annex A describes the TPs for subgroup 1.1 (shown in bold).

– Group 1: HFS sender (HFSS)
  - **Group 1.1: Web services interoperability (WSI)**
    - **Subgroup 1.1.1: Basic profile (BP)**
    - **Subgroup 1.1.2: Basic security profile (BSP)**
    - **Subgroup 1.1.3: Reliable messaging (RM)**
  - Group 1.2: Simple object access protocol (SOAP)
    - Subgroup 1.2.1: SOAP headers (HEAD)
  - Group 1.3: Audit trail and node authentication (ATNA)
    - Subgroup 1.3.1: General (GEN)
    - Subgroup 1.3.2: PCD-01 (PCD-01)
    - Subgroup 1.3.3: Consent Management (CM)
  - Group 1.4: PCD-01 HL7 messages (PCD-01-DATA)
    - Subgroup 1.4.1: General (GEN)
    - Subgroup 1.4.2: Design guidelines (DG)
    - Subgroup 1.4.3: Pulse oximeter (PO)
    - Subgroup 1.4.4: Blood pressure monitor (BPM)
    - Subgroup 1.4.5: Thermometer (TH)
    - Subgroup 1.4.6: Weighing scales (WEG)
    - Subgroup 1.4.7: Glucose meter (GL)
    - Subgroup 1.4.8: Cardiovascular fitness and activity monitor (CV)
    - Subgroup 1.4.9: Strength fitness equipment (ST)
    - Subgroup 1.4.10: Independent living activity hub (HUB)
    - Subgroup 1.4.11: Adherence monitor (AM)
    - Subgroup 1.4.12: Peak expiratory flow monitor (PF)
    - Subgroup 1.4.13: Body composition analyser (BCA)
    - Subgroup 1.4.14: Basic electrocardiograph (ECG)
    - Subgroup 1.4.15: International normalized ratio (INR)
    - Subgroup 1.4.16: Sleep apnoea breathing therapy equipment (SABTE)
    - Subgroup 1.4.17: Insulin pump (IP)
    - Subgroup 1.4.18: Continuous glucose monitor (CGM)
  - Group 1.5: Consent Management (CM)
    - Subgroup 1.5.1: HFS XDR transaction (TRANS)

        - Subgroup 1.5.2: HFS metadata validation (META)
        - Subgroup 1.5.3: HFS consent directive validation (CDV)
    - Group 1.6: hData Observation Upload (HDATA)
        - Subgroup 1.6.1: General (GEN)
    - Group 1.7: Questionnaires (QUE)
        - Subgroup 1.7.1: General (GEN)
        - Subgroup 1.7.2: CDA validation (CDA)
- Group 2: HFS receiver (HFSR)
    - Group 2.1: Web service interoperability (WSI)
        - Subgroup 2.1.1: Basic profile (BP)
        - Subgroup 2.1.2: Basic security profile (BSP)
        - Subgroup 2.1.3: Reliable messaging (RM)
    - Group 2.2: SOAP (SOAP)
        - Subgroup 2.2.1: SOAP headers (HEAD)
    - Group 2.3: Audit (ATNA)
        - Subgroup 2.3.1: General (GEN)
        - Subgroup 2.3.2: PCD-01 (PCD-01)
        - Subgroup 2.3.3: Consent Management (CM)
    - Group 2.4: PCD-01 HL7 messages (PCD-01-DATA)
        - Subgroup 2.4.1: General (GEN)
        - Subgroup 2.4.2: Design guidelines (DG)
        - Subgroup 2.4.3: Pulse oximeter (PO)
        - Subgroup 2.4.4: Blood pressure monitor (BPM)
        - Subgroup 2.4.5: Thermometer (TH)
        - Subgroup 2.4.6: Weighing scales (WEG)
        - Subgroup 2.4.7: Glucose meter (GL)
        - Subgroup 2.4.8: Cardiovascular fitness and activity monitor (CV)
        - Subgroup 2.4.9: Strength fitness equipment (ST)
        - Subgroup 2.4.10: Independent living activity hub (HUB)
        - Subgroup 2.4.11: Adherence monitor (AM)
        - Subgroup 2.4.12: Peak expiratory flow monitor (PF)
        - Subgroup 2.4.13: Body composition analyser (BCA)
        - Subgroup 2.4.14: Basic electrocardiograph (ECG)
        - Subgroup 2.4.15: International normalized ratio (INR)
        - Subgroup 2.4.16: Sleep apnoea breathing therapy equipment (SABTE)
        - Subgroup 2.4.17: Insulin pump (IP)
        - Subgroup 2.4.18: Continuous glucose monitor (CGM)
    - Group 2.5: Consent Management (CM)
        - Subgroup 2.5.1: HFS XDR transaction (TRANS)
        - Subgroup 2.5.2: HFS service validation (SER)
    - Group 2.6: hData Observation Upload (HDATA)

- Subgroup 2.6.1: General (GEN)
- Subgroup 2.6.2: hData record format (HRF)

- Group 2.7: Questionnaires (QUE)
  - Subgroup 2.7.1: General (GEN)
  - Subgroup 2.7.2: CDA validation (CDA)
  - Subgroup 2.7.3: hData record format (HRF)

7 Electronic attachment

The protocol implementation conformance statements (PICS) and the protocol implementation extra information for testing (PIXIT) required for the implementation of Annex A can be downloaded from http://handle.itu.int/11.1002/2000/12067.

In the electronic attachment, letters "C" and "I" in the column labelled "Mandatory" are used to distinguish between "PICS" and "PIXIT" respectively during testing. If the cell is empty, the corresponding PICS is "independent". If the field contains a "C", the corresponding PICS is dependent on other PICS, and the logical expression is detailed in the "SCR_Expression" field. The static conformance review (SCR) is used in the test tool to assert whether the PICS selection is consistent.

Annex A

Test purposes

(This annex forms an integral part of this Recommendation.)

A.1 TP definition conventions

The test purposes (TPs) are defined according to the following rules:

– **TP Id**: This is a unique identifier (TP/<TT>/<DUT>/<GR>/<SGR>/<XX> – <NNN>). It is specified according to the naming convention defined below:
  - Each test purpose identifier is introduced by the prefix "TP".
  - <TT>: This is the test tool that will be used in the test case.
    – HFS: Health & Fitness Services Interface
  - <DUT>: This is the device under test.
    – SEN: HFS sender
    – REC: HFS receiver
  - <GR>: This identifies a group of test cases.
  - <SGR>: This identifies a subgroup of test cases.
  - <XX>: This identifies the type of testing.
    – BV: Valid behaviour test
    – BI: Invalid behaviour test
  - <NNN>: This is a sequential number that identifies the test purpose.

– **TP label**: This is the TP's title.

– **Coverage**: This contains the specification reference and clause to be checked by the TP.
  - Spec: This indicates the earliest version of the specification from which the testable items to be checked by the TP were included.
  - Testable item: This contains testable items to be checked by the TP.

– **Test purpose**: This is a description of the requirements to be tested.

– **Applicability**: This contains the PICS items that define if the test case is applicable or not for a specific device. When a TP contains an "ALL" in this field it means that it applies to the device under test within that scope of the test (specialization, transport used, etc.).

– **Other PICS**: This contains additional PICS items (apart from the PICS specified in the Applicability row) which are used within the test case implementation and can modify the final verdict. When this row is empty, it means that only the PICS specified in the Applicability row are used within the test case implementation.

– **Initial condition**: This indicates the state to which the DUT needs to be moved at the beginning of TC execution.

– **Test procedure**: This describes the steps to be followed in order to execute the test case.

– **Pass/Fail criteria**: This provides criteria to decide whether the DUT passes or fails the test case.

A.2 Subgroup 1.1.1 – Basic profile (BP)

<table>
<tr><td colspan="2">TP Id</td><td colspan="3">TP/HFS/SEN/WSI/BP/BV-000</td></tr>
<tr><td colspan="2">TP label</td><td colspan="3">SOAP Envelope Structure</td></tr>
<tr><td rowspan="4">Coverage</td><td>Spec</td><td colspan="3">[OASIS/WS-I BP]</td></tr>
<tr><td rowspan="3">Testable items</td><td>BP-R9980; M</td><td>BP-R9981; M</td><td>BP-R1014; M</td></tr>
<tr><td>BP-R1008; M</td><td>BP-R1009; M</td><td>BP-R1033; R</td></tr>
<tr><td>BP-R1032; M</td><td></td><td></td></tr>
<tr><td colspan="2">Test purpose</td><td colspan="3">Check that:
An Envelope must conform to the structure specified in SOAP1.2 Section 5.1, “SOAP Envelope”
[AND]
an Envelope must have exactly zero or one child elements of the soap:Body element
[AND]
the children of the soap:body element in an Envelope must be namespace qualified
[AND]
An Envelope must not contain a Document Type Declaration (DTD) or Processing Instructions
[AND]
an Envelope should not contain the namespace declaration xmlns:xml=”http//www.w3.org/XML/1998/namespace
[AND]
the soap:envelope, soap:header and soap:body elements must not have attributes in the namespace “http://schemas.xmlsoap.org/soap/envelope/”</td></tr>
<tr><td colspan="2">Applicability</td><td colspan="3">C_SEN_000 AND C_SEN_GEN_003</td></tr>
<tr><td colspan="2">Other PICS</td><td colspan="3"></td></tr>
<tr><td colspan="2">Initial condition</td><td colspan="3">The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.</td></tr>
<tr><td colspan="2">Test procedure</td><td colspan="3">1. The HFS sender under test sends the SOAP message to the HFS receiver.
2. Check that the captured message has the following structure:
<soap:Envelope <i>'namespace'</i>>
<soap:Header>
...
</soap:Header>
<soap:Body>
<i>Here are the children of soap:Envelope</i>
</soap:Body>
</soap:Envelope>
where soap:Header is optional and it is recommended that the namespace is not http://www.w3.org/XML/1998/namespace.</td></tr>
<tr><td colspan="2">Pass/Fail criteria</td><td colspan="3">Check that:
• The message has, in this order, an envelope, an optional header and a body.
• The namespaces that appear in the soap message are qualified.
• Soap:envelope, soap:header and soap:body do not have attributes in the namespace "http://schemas.xmlsoap.org/soap/envelope/".
• There is no DTD or processing instructions in the envelope.
• The SOAP envelope's namespace is "http://www.w3.org/2003/05/soap-envelope" to</td></tr>
</table>

	support SOAP 1.2 [b-SOAP 1.2].
Notes	BP-R2201 and BP-R2210 imply that there may be at most one child element of the soap:Body. The referenced errata, NE05, would not be allowed by Continua (not compliant with the WS-I Profile).

TP Id		TP/HFS/SEN/WSI/BP/BV-001		
TP label		SOAP encodingStyle Attribute		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R1005; M	BP-R1006; M	BP-R1007; M
Test purpose		Check that: An Envelope must not contain soap:encodingStyle attributes on any of the elements whose namespace name is “http://schemas.xmlsoap.org/soap/envelope/” [AND] an Envelope must not contain soap:encodingStyle attributes on any element that is a child of soap:body. [AND] an Envelope described in an rpc-literal binding must not contain soap:encodingStyle attribute on any element that is a grandchild of soap:Body		
Applicability		C_SEN_000 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		
Test procedure		1. Make the HFS sender under test send a SOAP message. 2. Check within the captured message: a. If the soap:encodingStyle attribute is present, that the envelope contains: ❑ a namespace which is not "http://schemas.xmlsoap.org/soap/envelope/" ❑ an element that is not a child of soap:Body. ❑ If an rpc-literal binding is used, check that the element is not a grandchild of soap:body.		
Pass/Fail criteria		If present, the soap:encodingStyle attribute is as specified within the test procedure above.		
Notes				

TP Id		TP/HFS/SEN/WSI/BP/BV-002		
TP label		Use of SOAP in HTTP		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R1132; M	BP-R1140; M	
Test purpose		Check that: A HTTP request message must use the HTTP POST method. [AND] A Message shall be sent using HTTP/1.1		
Applicability		C_SEN_000 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled and the HFS sender under test is ready to send an HTTP request.		

Test procedure	1. Make the HFS sender under test send a message to the simulated HFS receiver using the HTTP protocol. 2. Check in the HTTP header of the captured message that: a. the HTTP version is 1.1 b. POST method is used.
Pass/Fail criteria	Check that all values are as specified in the HTTP header.
Notes	

TP Id		TP/HFS/SEN/WSI/BP/BV-003		
TP label		HTTP Status Codes		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R1131; O		
Test purpose		Check that: A consumer may automatically redirect a request when it encounters a “307 Temporary Redirect” HTTP status code in a response		
Applicability		C_SEN_000 AND C_SEN_GEN_003		
Other PICS		C_SEN_WSI_001		
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has an HTTP request ready to be sent to the respective service according to its needs.		
Test procedure		1. Make the HFS sender under test send an HTTP request to the HFS receiver. 2. The simulated HFS receiver responds with "307 Temporary Redirect" as the status code. 3. If C_SEN_WSI_001=TRUE, the HFS sender redirects the request, or else the HFS sender does not redirect the request.		
Pass/Fail criteria		If C_SEN_WSI_001=TRUE, the HFS sender redirects the request to the http address indicated in the "307 Temporary Redirect" HTTP response.		
Notes				

TP Id		TP/HFS/SEN/WSI/BP/BV-004		
TP label		Messages using wsdl descriptions		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R2211; M	BP-R2212; M	BP-R2213; M
		BP-R2214; M		
Test purpose		Check that: An Envelope described with an rpc-literal binding must not have the xsi:nil attribute with a value of “1” or “true” on the part accessors [AND] an Envelope must contain exactly one part accessor element for each of the wsdl:parts in the same wsdl:message that are referred to by its soapbind:body element(s) [AND] in a doc-literal description where the value of the parts attribute of soapbind:body is an empty string, the corresponding envelope must have no element content in the soap:Body element [AND] in a rpc-literal description where the value of the parts attribute of soapbind:body is an empty string, the corresponding envelope must have no part accessor elements.		

Applicability	C_SEN_000 AND C_SEN_GEN_003
Other PICS	C_SEN_WSI_021
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Wait until the HFS sender under test sends a SOAP message or, if necessary, force it to send a SOAP message. 2. Take the WSDL description of the web service using its URL and check the soap envelope of the captured message: If an rpc-literal binding is used: a. If the soapbind:body element of the description is an empty string, there is no part accessor elements. b. If the soapbind:body element of the description is not empty, check that the part accessor of the envelope is present and that there is no xsi:nil attribute with a value of "1" or "true". If doc-literal binding is used: a. If the value of the parts attribute of soapbind:body is an empty string, the envelope does not have element content in soap:Body element.
Pass/Fail criteria	Check that the envelope is as specified in step 2.
Notes	

TP Id		TP/HFS/SEN/WSI/BP/BV-005		
TP label		Port Types		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R2301; M		
Test purpose		Check that: The order of the elements in the soap:Body of an envelope must be the same as that of the wsdl:parts in the wsdl:message that describes it for each of the wsdl:part elements bound to the envelope's corresponding soapbind:body element		
Applicability		C_SEN_000 AND C_SEN_GEN_003		
Other PICS		C_SEN_WSI_021		
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		
Test procedure		1. Wait until the HFS sender under test sends a SOAP message or, if necessary, force it to send a SOAP message. 2. Take the WSDL description of the web service using its URL and check the wsdl:parts elements in the wsdl:message. 3. Compare their order with the soap:Body elements order.		
Pass/Fail criteria		In step 3, check that the order of the wsdl:parts are the same as the order of the elements in the soap:Body.		
Notes				

TP Id		TP/HFS/SEN/WSI/BP/BV-006		
TP label		SOAP Binding		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R2742; O	BP-R2743; O	

Test purpose	Check that: An envelope may contain fault with a detail element that is not described by a soapbind:fault element in the corresponding WSDL description [AND] An envelope may contain the details of a header processing related fault in a SOAP header block that is not described by a soapbind:headerfault element in the corresponding WSDL description
Applicability	C_SEN_000 AND C_SEN_WSI_034 AND C_SEN_GEN_003
Other PICS	C_SEN_WSI_021
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Wait until the HFS sender under test sends a SOAP message or, if necessary, force it to send a SOAP message. 2. The simulated HFS receiver responds with a message that will cause that HFS sender to generate a fault. 3. The HFS sender under test sends a fault message. 4. Check the envelope's fault detail element and the SOAP header block's header processing fault.
Pass/Fail criteria	In step 2, verify that the detail element cannot be described by the soapbind:fault element of the WSDL description, and that the header block cannot be described by a soapbind:headerfault element of the WSDL description.
Notes	

TP Id		TP/HFS/SEN/WSI/BP/BV-006_B		
TP label		SOAP Binding 2		
Coverage	**Spec**	[OASIS/WS-I BP]		
	Testable items	BP-R2712; M	BP-R2735; M	BP-R2755; M
		BP-R2737; M	BP-R2738; M	BP-R2739; O
		BP-R2752; O	BP-R2753; O	
Test purpose		Check that: A document-literal binding must be serialized as an envelope with a soap:Body whose child element is an instance of the global element declaration referenced by the corresponding wsdl:message part [AND] An envelope described with an rpc-literal binding must place the part accessor elements for parameters and return value in no namespace [AND] The part accessor elements in a message described with an rpc-literal binding must have a local name of the same value as the name attribute of the corresponding wsdl:part element [AND] An envelope described with an rpc-literal binding must namespace qualify the descendants of part accessor elements for the parameters and the return value, as defined by the schema in which the part accessor types are defined [AND] An envelope must include all soapbind:headers specified on a wsdl:input or wsdl:output of a wsdl:operation of a wsdl:binding that describes it [AND] An Envelope may contain SOAP header blocks that are not described in the wsdl:binding that describes it		

	[AND] An envelope may contain more than one instance of each SOAP header block for each soapbind:header element in the appropriate child of soapbind:binding in the corresponding description [AND] An envelope containing SOAP header blocks that are not described in the appropriate wsdl:binding may have the mustUnderstand attribute on such SOAP header blocks set to '1'.
Applicability	C_SEN_000 AND C_SEN_GEN_003
Other PICS	C_SEN_WSI_021
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Wait until the HFS sender under test sends any SOAP message or, if necessary, force it to send any SOAP message. 2. Check the captured message.
Pass/Fail criteria	Look into the WSDL description of the web service and check: • in step 2: o if the SOAP header block is not described in the wsdl:binding, it may be present and it is optional that the mustUnderstand attribute is present and equal to "1", and that the envelope has more than one instance for each header block; o that all soapbind:headers specified in wsdl:input or wsdl:output of a wsdl:operation of a wsdl:binding are included in the envelope; o if an rpc-literal binding is used; that the part accessor of the envelope has a local name equal to the name of the attribute of the wsdl:part element; that it is not placed in a namespace; and that its descendants have a namespace qualified by the schema in which the part accessor types are defined; o if a doc-literal binding is used, that the child element of the soap:Body is an instance of the global element declaration referenced by the corresponding wsdl:message part.
Notes	

A.3 Subgroup 1.1.2 – Basic security profile (BSP)

TP Id		TP/HFS/SEN/WSI/BSP/BV-000		
TP label		TLS Ciphersuites		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-322; R	BSP-323; R	
	Spec	[ITU-T H.812]		
	Testable items	SecGuidelines2; M		
Test purpose		Check that: As the AES encryption algorithm is intended to supersede the 3DES algorithm, it is recommended that TLS-capable implementations implement TLS_RSA_WITH_AES_128_CBC_SHA or the FIPS equivalent [AND] The ciphersuites defined in the TLS specifications that use anonymous Diffie-Hellman (i.e. those that have DH_anon in their symbolic name) are vulnerable to man-in-the-middle attacks. It is also recommended that ciphersuites that include MD5 (i.e. those that have MD5 in their symbolic name) be avoided, due to known security weaknesses of the MD5 algorithm. It is recommended that such ciphersuites be avoided. The Profile recommends against the use of the following ciphersuites due to their lack of confidentiality services: - TLS_RSA_WITH_NULL_SHA		

	- TLS_RSA_WITH_NULL_MD5 It is also recommended that ciphersuites that use 40 or 56 bit keys be avoided, due to their relative ease of compromise through brute-force attack. [AND] Continua HFS client and service components shall support AES cipher as specified in RFC 3268.
Applicability	C_SEN_000 AND C_SEN_GEN_003
Other PICS	C_SEN_WSI_002, C_SEN_WSI_027, C_SEN_WSI_028, C_SEN_WSI_029, C_SEN_WSI_030
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. If an instance is FIPS compliant (C_SEN_WSI_002=true): a. Load the simulated HFS receiver supporting TLS_RSA_FIPS_WITH_AES_128_CBC_SHA. b. Make the HFS sender under test establish a TLS connection. c. Check in the TLS handshake that the HFS sender under test SHOULD not support: ❑ any ciphersuites with an DH_anon in their symbolic name ❑ any ciphersuites with a MD5 in their symbolic name ❑ any of the following ciphersuites: • TLS_RSA_WITH_NULL_SHA • TLS_RSA_WITH_NULL_MD5 ❑ any ciphersuites that use 40 or 56 bit keys. d. Check that the HFS sender under test supports TLS_RSA_FIPS_WITH_AES_128_CBC_SHA e. Close the connection. 2. If an instance is not FIPS compliant (C_SEN_WSI_002=false): a. Load the simulated HFS receiver supporting TLS_RSA_WITH_AES_128_CBC_SHA. b. Make the HFS sender under test establish a TLS connection. c. Check in the TLS handshake that the HFS sender under test does not support (these are recommendations only): ❑ any ciphersuites with an DH_anon in their symbolic name ❑ any ciphersuites with a MD5 in their symbolic name ❑ any of the following ciphersuites: • TLS_RSA_WITH_NULL_SHA • TLS_RSA_WITH_NULL_MD5 ❑ any ciphersuites that use 40 or 56 bit keys. d. Check that the HFS sender under test supports: TLS_RSA_WITH_AES_128_CBC_SHA.
Pass/Fail criteria	• If C_SEN_WSI_002 is supported, the HFS sender under test must support TLS_RSA_FIPS_WITH_AES_128_CBC_SHA. • If C_SEN_WSI_002 is not supported, the HFS sender under test must support TLS_RSA_WITH_AES_128_CBC_SHA. • The ciphersuites supported must match with these PICS: C_SEN_WSI_027, C_SEN_WSI_028, C_SEN_WSI_029, C_SEN_WSI_030.
Notes	
TP Id	TP/HFS/SEN/WSI/BSP/BV-001

TP label		Security Policy		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3105; O		
Test purpose		Check that: An HFS sender may agree in an out of band fashion with an HFS receiver on required and allowed signed and/or encrypted message content and security tokens		
Applicability		C_SEN_000 AND C_SEN_WSI_003 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled with many different services. The HFS sender under test and the simulated HFS receiver have never been partners in a message exchange.		
Test procedure		1. Make the HFS sender under test send its supported configuration to the HFS receiver, including supported encryption and/or signatures and security tokens. 2. The simulated HFS receiver waits for a SOAP message from the HFS sender. 3. The simulated HFS receiver checks the received message, ensuring that the HFS sender agrees or disagrees in an out of band fashion with the HFS receiver.		
Pass/Fail criteria		Step 3 is achieved.		
Notes		This is WS-Trust negotiation.		

TP Id		TP/HFS/SEN/WSI/BSP/BV-003		
TP label		Basic Profile Clarification		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R5801; M	BSP-R5805; M	BSP-R5813; M
Test purpose		Check that: bp11:R2301 must be true after any SOAP Message Security has been reversed for the Envelope. Bp11:R2301 states “the order of the elements in the soap:body of an Envelope must be the same as that of the wsdl:parts in the wsdl:message that describes it”. [AND] bp11:R2712 must be true after any SOAP Message Security has been reversed for the Envelope. Bp11:R2712 states “A document-literal binding must be serialized as an Envelope with a soap:body whose child element is an instance of the global element declaration referenced by the corresponding wsdl:message part” [AND] With respect to bp11:R2738 verification of an Envelope must occur after SOAP Message Security has been reversed. Bp11:R2738 states “an Envelope must include all soapbind:headers specified on a wsdl:input or wsdl:output of a wsdl:operation of a wsdl:binding that describes it”.		
Applicability		C_SEN_000 AND C_SEN_WSI_003 AND C_SEN_GEN_003		
Other PICS		C_SEN_WSI_021		
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		
Test procedure		1. Make the HFS sender under test send a SOAP message using security. 2. As the simulated HFS receiver knows its description (wsdl), after reversing the SOAP message security, check that: a. The order of the elements in the soap:body is the same as the wsdl:parts in the wsdl:message. b. The envelope includes all soapbind:headers specified on a wsdl:input or wsdl:output of a wsdl:operation of a wsdl:binding.		

	c. If doc-literal binding is used, it is serialized as an envelope with a soap:Body whose child element is an instance of the global element declaration referenced by the corresponding wsdl:message part.
Pass/Fail criteria	All steps are as specified within the test procedure above.
Notes	"Reversing SOAP Message Security" means removing the various impacts of applying "SOAP Message Security" that may have been applied since the MESSAGE (BP1.0) or ENVELOPE (BP 1.1) was originally created for that recipient according to the BP. This may mean decrypting relevant portions of the XML or removing XML signature elements or making other reverse transformations as appropriate to the aspects of SOAP message security that were applied in the specific circumstance.

TP Id		TP/HFS/SEN/WSI/BSP/BV-005		
TP label		Timestamp element		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3227; M	BSP-R3203; M	BSP-R3224; R
		BSP-R3221; M	BSP-R3222; M	BSP-R3220; R
		BSP-R3229; R	BSP-R3213; M	BSP-R3215; M
		BSP-R3225; M	BSP-R3226; M	BSP-R3217; M
		BSP-R3223; M		
Test purpose		Check that: A SECURITY_HEADER must not contain more than one Timestamp [AND] A Timestamp must contain exactly one Created [AND] Any Timestamp must not contain more than one Expires [AND] Any Timestamp containing an Expires must contain a Created that precedes its sibling Expires [AND] Any Timestamp must not contain anything other than Created or Expires elements [AND] Any Created should not contain a seconds value with more than three digits to the right of the decimal (milliseconds). [AND] Any Expires should not contain a seconds value with more than three digits to the right of the decimal (milliseconds). [AND] Any Created containing second values must specify seconds values less than 60 [AND] Any Expires containing second values must specify seconds values less than 60 [AND] Any Created must not include a ValueType attribute [AND] Any Expires must not include a ValueType attribute [AND] Any Created must contain time values in UTC format as specified by the XML Schema type (dateTime). [AND]		

	Any Expires must contain time values in UTC format as specified by the XML Schema type (dateTime).
Applicability	C_SEN_000 AND C_SEN_WSI_004 AND C_SEN_GEN_003
Other PICS	C_SEN_WSI_021
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Make the HFS sender under test send a SOAP message using a Timestamp element. 2. Check in the captured message that: a. Timestamp is present and there is only one. For example: <wsu:Timestamp wsu:Id="*timestamp*"> <wsu:Created>*2001-09-13T08:42:00Z*</wsu:Created> <wsu:Expires>*2001-10-13T09:00:00Z*</wsu:Expires> </wsu:Timestamp> b. Only one Created element is present and inside it: ❑ ValueType attribute is not included ❑ UTC format is used in time values ❑ seconds values are less than 60 and its decimal values are recommended to be less than 3 digits to the right. c. If the Expires element is present, only one, it comes after the Created element and: ❑ ValueType attribute is not included ❑ UTC format is used in time values ❑ seconds values are less than 60 and its decimal values are recommended to be less than 3 digits to the right.
Pass/Fail criteria	The elements in step 2 are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/BSP/BV-006		
TP label		Security Token References – Direct References		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3061; M	BSP-R3057; M	BSP-R3064; M
		BSP-R3059; M	BSP-R3058; M	BSP-R3062; M
		BSP-R3027; M	BSP-R3211; M	
Test purpose		Check that: A SECURITY_TOKEN_REFERENCE must provide exactly one token reference [AND] Any STR_REFERENCE must not reference a SECURITY_TOKEN_REFERENCE [AND] Any STR_REFERENCE must not reference an STR_EMBEDDED [AND] Any STR_REFERENCE must specify a ValueType attribute [AND] Any STR_REFERENCE ValueType attribute must contain a value for the referenced SECURITY_TOKEN specified by the corresponding security token profile. [AND] Any STR_REFERENCE must specify a URI attribute		

	[AND] Any SECURITY_TOKEN_REFERENCE must not contain an STR_KEY_NAME [AND] Any SECURITY_TOKEN_REFERENCE must not reference a ds:KeyInfo element
Applicability	C_SEN_000 AND C_SEN_WSI_016 AND C_SEN_GEN_003
Other PICS	
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Make the HFS sender under test send a SOAP message using a security token reference (STR) with an STR_Reference. <wsse:SecurityTokenReference wsu:Id="..."> <wsse:Reference URI="..." ValueType="..."/> </wsse:SecurityTokenReference> 2. Check in the captured message that: a. There is only one STR_Reference within the SECURITY_TOKEN_REFERENCE. b. STR_Reference does not reference another SECURITY_TOKEN_REFERENCE or an STR_Embedded. c. URI Attribute is present. d. ValueType attribute is present and it contains a value for the referenced security token specified by the corresponding security token profile (e.g., X.509 certificate token). e. SECURITY_TOKEN_REFERENCE does not contain an STR_KEY_NAME and does not reference a ds:KeyInfo element.
Pass/Fail criteria	Check that SECURITY_TOKEN_REFERENCE is as specified in steps 1 and 2.
Notes	

TP Id		TP/HFS/SEN/WSI/BSP/BV-007		
TP label		Security Token References – Key Identifier		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3054; M	BSP-R3063; M	BSP-R3070; M
		BSP-R3071; M		
Test purpose		Check that: Any STR_KEY_IDENTIFIER must specify a ValueType attribute [AND] Any STR_KEY_IDENTIFIER ValueType attribute must contain a value specified within the security token profile associated with the referenced SECURITY_TOKEN [AND] Any STR_KEY_IDENTIFIER that refers to a SECURITY_TOKEN other than a SAML_TOKEN must specify an EncodingType attribute [AND] Any STR_KEY_IDENTIFIER EncodingType attribute must have a value of "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary".		
Applicability		C_SEN_000 AND C_SEN_WSI_017 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		

Test procedure	1. Make the HFS sender under test send a SOAP message using a security token reference (STR) with a key identifier reference: <wsse:SecurityTokenReference> <wsse:KeyIdentifier wsu:Id="..." ValueType="..." EncodingType="...">...</wsse:KeyIdentifier> </wsse:SecurityTokenReference> 2. Check in the captured message that: a. ValueType is present and contains a value specified within the security token profile associated with the referenced security token. b. If an SAML token is referenced, the encodingType attribute is not present. c. If the referenced token is different from the SAML token, the encodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary".
Pass/Fail criteria	In step 2, attributes are as specified.
Notes	

TP Id		TP/HFS/SEN/WSI/BSP/BV-008		
TP label		Security Token References – Embedded References		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3060; M	BSP-R3025; M	BSP-R3056; M
Test purpose		Check that: Any STR_EMBEDDED must contain only a single child element which is an INTERNAL_SECURITY_TOKEN [AND] Any INTERNAL_SECURITY_TOKEN contained in an STR_EMBEDDED must be in the same format as if it were a child of a SECURITY_HEADER [AND] Any STR_EMBEDDED must not contain a wsse:SecurityTokenReference child element		
Applicability		C_SEN_000 AND C_SEN_WSI_018 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		
Test procedure		1. Make the HFS sender under test send a SOAP message using a security token reference (STR) with an embedded reference: <wsse:SecurityTokenReference> <wsse:Embedded wsu:Id="...">...</wsse:Embedded> </wsse:SecurityTokenReference> 2. Check in the captured message that: a. STR_Embedded has only one child element that is an internal security token, and it is in the same format as if it were a child of a security header. b. STR_Embedded does not contain a wsse:SecurityTokenReference child element.		

Pass/Fail criteria	In step 2, "Security Token Reference Embedded" are as specified.
Notes	An internal token reference is a reference to a token that is contained in the same message. An example of an incorrect and a correct format are: INCORRECT: <!-- This example is incorrect because the wsse:Embedded element carries the data for the X.509 certificate directly rather than as a wsse:BinarySecurityToken element --> <wsse:SecurityTokenReference> <wsse:Embedded wsu:Id="SomeCert"> lui+Jy4WYKGJW5xM3aHnLxOpGVIpzSg4V486hHFe7sHET/uxxVBovT7JV1A2RnWSWkXm9jAEdsm/... </wsse:Embedded> </wsse:SecurityTokenReference> CORRECT: <wsse:SecurityTokenReference> <wsse:Embedded wsu:Id="TheEmbeddedElementAroundSomeCert"> <wsse:BinarySecurityToken wsu:Id='SomeCert' ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0#X509v3" EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0#Base64Binary"> lui+Jy4WYKGJW5xM3aHnLxOpGVIpzSg4V486hHFe7sHET/uxxVBovT7JV1A2RnWSWkXm9jAEdsm/... </wsse:BinarySecurityToken> </wsse:Embedded> </wsse:SecurityTokenReference>

TP Id		TP/HFS/SEN/WSI/BSP/BV-009		
TP label		Security Token References – Internal References		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3022; M	BSP-R3023; M	BSP-R5204; M
		BSP-R5205; M	BSP-R3067; M	
Test purpose		Check that: Any SECURITY_TOKEN_REFERENCE that references an INTERNAL_SECURITY_TOKEN which has a wsu:Id attribute must contain an STR_REFERENCE or STR_EMBEDDED [AND] Any SECURITY_TOKEN_REFERENCE that references an INTERNAL_SECURITY_TOKEN that is referenced several times should contain an STR_REFERENCE rather than an STR_EMBEDDED [AND] Any STR_REFERENCE to an INTERNAL_SECURITY_TOKEN having an ID attribute must contain a URI attribute with a Shorthand XPointer value [AND] Any INTERNAL_SECURITY_TOKEN that is not contained in an STR_EMBEDDED must precede all SECURITY_TOKEN_REFERENCE elements that reference it in the SOAP Envelope [AND] Any STR_REFERENCE that is a descendant of an ENCRYPTED_DATA must not use a Shorthand XPointer to refer to an INTERNAL_SECURITY_TOKEN located in a SECURITY_HEADER other than the SECURITY_HEADER containing a reference		

	(EK_REFERENCE_LIST or an ENC_REFERENCE_LIST) to the ENCRYPTED_DATA
Applicability	C_SEN_000 AND C_SEN_WSI_019 AND C_SEN_GEN_003
Other PICS	
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Make the HFS sender under test send a SOAP message including a SecurityTokenReference with an internal reference. 2. Check in the captured message that: a. The SECURITY_TOKEN_REFERENCE references an internal security token. b. The SECURITY_TOKEN_REFERENCE contains an STR_Reference or STR_Embedded. It is recommended to be an STR_Reference. c. The STR_Reference to an INTERNAL_SECURITY_TOKEN which has an ID attribute contains a URI attribute with a shorthand XPointer value. d. The INTERNAL_SECURITY_TOKEN precedes all SECURITY_TOKEN_REFERENCE elements that reference it in the SOAP envelope.
Pass/Fail criteria	References are as specified within the test procedure above.
Notes	The internal token reference is a reference to a token that is contained in the same message.

TP Id		TP/HFS/SEN/WSI/BSP/BV-010		
TP label		Security Token References – External References		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R3024; M		
Test purpose		Check that: Any EXTERNAL_TOKEN_REFERENCE that can use an STR_REFERENCE must contain an STR_REFERENCE		
Applicability		C_SEN_000 AND C_SEN_WSI_020 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.		
Test procedure		1. Make the HFS sender under test send a SOAP message including a SecurityTokenReference with an external reference. 2. Check in the text file that: a. It is recommended that the external token reference contain an STR_Reference.		
Pass/Fail criteria		References are as specified within the test procedure above.		
Notes		The external token reference is a reference to a token that is not contained in the same message.		

TP Id		TP/HFS/SEN/WSI/BSP/BV-023		
TP label		SAML Token		
Coverage	**Spec**	[OASIS/WS-I BSP]		
	Testable items	BSP-R6601; M	BSP-R6602; M	BSP-R6609; M
		BSP-R6603; M	BSP-R6604; M	BSP-R6605; M
		BSP-R6606: M	BSP-R6607; M	BSP-R6608; M

Test purpose	Check that: Any SAML_SC_KEY_INFO must not contain a reference to a SAML_TOKEN [AND] Any STR_KEY_IDENTIFIER that references a INTERNAL_SAML_TOKEN must include a ValueType attribute [AND] Any STR_KEY_IDENTIFIER that references a EXTERNAL_SAML_TOKEN must include a ValueType attribute [AND] Any STR_KEY_IDENTIFIER ValueType attribute that references SAML_TOKEN must have a value of "http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.1#SAMLID" [AND] Any STR_KEY_IDENTIFIER that references a SAML_TOKEN must not include an EncodingType attribute [AND] Any STR_KEY_IDENTIFIER that references a SAML_TOKEN must have a value encoded as an xs:string [AND] Any SECURITY_TOKEN_REFERENCE that references an EXTERNAL_SAML_TOKEN must contain a SAML_AUTHORITY_BINDING [AND] Any AuthorityKind attribute of a SAML_AUTHORITY_BINDING must have a value of saml:AssertionIdReference [AND] Any SECURITY_TOKEN_REFERENCE that references an INTERNAL_SAML_TOKEN must not contain a SAML_AUTHORITY_BINDING
Applicability	C_SEN_000 AND C_SEN_GEN_003
Other PICS	
Initial condition	The simulated HFS receiver has a WebService enabled with many different services and the HFS sender under test has a SOAP message ready to be sent to the respective service according to its needs.
Test procedure	1. Make the HFS sender under test send a SOAP message using an SAML token. 2. Check in the captured message that the expected saml:Assertion element confirms that: a. SAML KeyInfo does not contain a reference to an SAML token. b. In an STR KeyIdentifier that references an SAML token: ❑ EncodingType attribute is not present. ❑ ValueType="http://docs.oasis-open.org/wss/oasis-wss-saml-token-profile-1.1#SAMLID". ❑ The Value encoded is an xs:string. c. If a security token reference references an external SAML token: ❑ saml:AuthorityBinding element is present ❑ AuthorityKind = Value of saml:AssertionIdReference.
Pass/Fail criteria	The SAML token element is as specified within the test procedure above.
Notes	

A.4 Subgroup 1.1.3 – Reliable messaging (RM)

TP Id	TP/HFS/SEN/WSI/RM/BV-000
TP label	Protocol Preconditions

Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	Namespace; M	ProtocolPrec 2; M	
Test purpose		Check that: The XML namespace URI that MUST be used by implementations of this specification is: http://docs.oasis-open.org/ws-rx/wsrm/200702 [AND] The RM Source MUST have successfully created a Sequence with the RM Destination		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "None" sequence state.		
Test procedure		1. The HFS sender under test sends a CreateSequence message with an offer element to the HFS receiver. 2. The simulated HFS receiver responds with a CreateSequenceResponse message accepting the offer. 3. The HFS sender sends a Sequence message. 4. The HFS receiver responds with its Sequence message and a SequenceAcknowledgement element. 5. The HFS sender sends a SequenceAcknowledgement element.		
Pass/Fail criteria		Check that in every wsrm element its XML namespace is: xmlns:wsrm=" http://docs.oasis-open.org/ws-rx/wsrm/200702", and in step 1 the CreateSequence request is made.		
Notes				

TP Id		TP/HFS/SEN/WSI/RM/BV-001		
TP label		Delivery Assurances		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	DelivAssurance 4; C	DelivAssurance 7; C	
	Spec	[ITU-T H.812]		
	Testable items	CommonReq 2; O	CommonReq 3; R	
Test purpose		Check that: AtMostOnce assertion sets that each message is to be delivered at most once. The RM Source MAY retry transmission of unacknowledged messages, but is NOT REQUIRED to do so. [AND] The requirement on an RM Source using ExactlyOnce assertion is that it SHOULD retry transmission of every message sent by the Application Source until it receives an acknowledgement from the RM Destination [AND] Continua HFS client and service components may transmit messages from the Continua better QoS bin using a WS-ReliableMessaging sequence configured to use 'AtMostOnce' message delivery. [AND] Continua HFS client and service components should transmit messages from the Continua best QoS bin using a WS-ReliableMessaging sequence configured to use 'ExactlyOnce' message delivery.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND (C_SEN_WSI_023 OR C_SEN_WSI_024) AND C_SEN_GEN_003		

Other PICS	
Initial condition	The HFS sender under test and the simulated HFS receiver are in the "None" sequence state. The simulated HFS receiver is able to avoid the response to a CreateSequence message.
Test procedure	1. Make the HFS sender send a CreateSequence message. 2. The simulated HFS receiver does not respond to that message. 3. If C_SEN_WSI_023, the HFS sender may retry transmission. 4. If C_SEN_WSI_024, the HFS sender should retry transmission.
Pass/Fail criteria	All steps are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-003		
TP label		Consideration on the Use of "Piggy-Backing"		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	PiggyBack 1; O	PiggyBack 2; M	PiggyBack 3; R
Test purpose		Check that: Some RM Protocol Header Blocks MAY be added to messages that are targeted to the same Endpoint to which those headers are to be sent (a concept often referred to as "piggy-backing"), thus saving the overhead of an additional message exchange. [AND] Reference parameters MUST be considered when determining whether two EPRs are targeted to the same Endpoint [AND] In order to ensure optimal and successful processing of RM Sequences, endpoints that receive RM-related messages SHOULD be prepared to process RM Protocol Header Blocks that are included in any message it receives.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "None" sequence state.		
Test procedure		1. The HFS sender under test sends a CreateSequence message with an offer element. 2. The simulated HFS receiver responds with CreateSequenceResponse accepting the offer. 3. The HFS sender sends a Sequence message. 4. The HFS receiver responds with a SOAP message including a SequenceAcknowledgement header block and a Sequence header block (indicating that it is the last message). 5. The HFS sender responds including a SequenceAcknowledgement header block. ❑ If the SOAP message also contains a CloseSequence header block or any other header block (piggy-backing), all the header blocks will have the same EPR (endpoint reference). ❑ If not, any other header block is sent in the same SOAP message, the HFS sender under test sends a message for every other RM-element (not piggy-backing).		
Pass/Fail criteria		In step 5, If the HFS sender sends only one message with more than one header block (piggy-backing), the EPR is the same for every header block.		
Notes		An endpoint reference is made using a "wsa:To" element. The way to test that every header block is targeted to the same endpoint is that there is only one "wsa:To" element in the soap:header.		

TP Id		TP/HFS/SEN/WSI/RM/BV-004		
TP label		Sequence Creation		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	WSAddress 1; C	SeqCreation 1; M	SeqCreation 2; O
		SeqCreation 5; M	SeqCreation 7; M	SeqCreation 8; M
		SeqCreation 9; O	SeqCreation 10; M	SeqCreation 11; M
		SeqCreation 12; M	SeqCreation 14; O	SeqCreation 15; O
		SeqCreation 22; O		
Test purpose		Check that: When an Endpoint generates a message that carries an RM protocol element in the body of a SOAP envelope that Endpoint MUST include in that envelope a wsa:Action SOAP header block whose value is an IRI that is a concatenation of the WS-RM namespace URI, followed by a "/", followed by the value of the local name of the child element of the SOAP body. [AND] The RM Source MUST request creation of an outbound Sequence by sending a CreateSequence element in the body of a message to the RM Destination which in turn responds either with a message containing CreateSequenceResponse or a CreateSequenceRefused fault [AND] The RM Source MAY include an offer to create an inbound Sequence within the CreateSequence message. [AND] The RM Source MUST NOT send wsrm:CreateSequence element as a header block. [AND] The RM Source MUST include wsrm:AcksTo element in any CreateSequence message it sends. This element is of type wsa:EndpointReferenceType (as specified by WS-Addressing). It specifies the endpoint reference to which messages containing SequenceAcknowledgement header blocks and faults related to the created Sequence are to be sent, unless otherwise noted in this specification [AND] Implementations MUST NOT use an endpoint reference in the AcksTo element that would prevent the sending of Sequence Acknowledgements back to the RM Source. [AND] wsrm:Expires element, if present, of type xs:duration specifies the RM Source's requested duration for the Sequence. The RM Destination MAY either accept the requested duration or assign a lesser value of its choosing. A value of "PT0S" indicates that the Sequence will never expire. Absence of the element indicates an implied value of "PT0S" [AND] The RM Source MUST set the value of wsrm:Identifier element to an absolute URI (conformant with RFC3986) that uniquely identifies the offered Sequence [AND] An RM Source MUST include wsrm:Endpoint element, of type wsa:EndpointReferenceType (as specified by WS-Addressing). This element specifies the endpoint reference to which Sequence Lifecycle Messages, Acknowledgement Requests, and fault messages related to the offered Sequence are to be sent. [AND] Implementations MUST NOT use an endpoint reference in the Endpoint element that would prevent the sending of Sequence Lifecycle Message, etc. [AND] wsrm:Expires element within wsrm:Offer, if present, of type xs:duration specifies the duration for the offered Sequence. A value of "PT0S" indicates that the offered Sequence will never expire. Absence of the element indicates an implied value of "PT0S"		

	[AND] wsrm:IncompleteSequenceBehaviour element, if present in wsrm:Offer element within wsrm:CreateSequence element, specifies the behavior that the destination will exhibit upon the closure or termination of an incomplete Sequence. For the purposes of defining the values used, the term "discard" refers to behavior equivalent to the Application Destination never processing a particular message. A value of “DiscardEntireSequence” indicates that the entire Sequence MUST be discarded if the Sequence is closed, or terminated, when there are one or more gaps in the final SequenceAcknowledgement. A value of “DiscardFollowingFirstGap” indicates that messages in the Sequence beyond the first gap MUST be discarded when there are one or more gaps in the final SequenceAcknowledgement. The default value of “NoDiscard” indicates that no acknowledged messages in the Sequence will be discarded. [AND] If a CreateSequenceResponse is returned without a child Accept in response to a CreateSequence that did contain a child Offer, then the RM Source MAY immediately reclaim any resources associated with the unused offered Sequence.
Applicability	C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003
Other PICS	
Initial condition	The HFS sender under test and the simulated HFS receiver are in the "None" sequence state.
Test procedure	1. Wail until the HFS sender under test sends a CreateSequence message. 2. Check that the captured message has the following properties: a. In the header block: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/CreateSequence. ❑ wsrm:CreateSequence is not present. b. In the body of the message: ❑ wsrm:AcksTo of type wsa:EndpointReferenceType is present and defines a valid endpoint. ❑ wsrm:Expires element, if present: • its type is xs:duration. ❑ If an offer element is present: • wsrm:IncompleteSequenceBehaviour element may be present. Possible values are: "discard", "DiscardEntireSequence", "DiscardFollowingFirstGap" and "NoDiscard". • wsrm:Identifier value is an absolute URI that uniquely identifies the offered Sequence. • wsrm:Expires element, if present, its type is xs:duration. • wsrm:Endpoint element is present and its type is wsa:EndpointReferenceType, and it defines a valid endpoint. 3. The simulated HFS receiver responds using a CreateSequenceResponse message without an accept element or a CreateSequenceRefused fault. 4. If an offer element is present: ❑ The HFS sender can reclaim the resources.
Pass/Fail criteria	All elements are as specified within the test procedure above.
Notes	

TP Id	TP/HFS/SEN/WSI/RM/BV-005
TP label	Closing a Sequence

<table>
<tr><td rowspan="4">Coverage</td><td>Spec</td><td colspan="3">[OASIS WS-I RM]</td></tr>
<tr><td rowspan="3">Testable items</td><td>WSAddress 1; C</td><td>SeqClosing 1; O</td><td>SeqClosing 2; M</td></tr>
<tr><td>SeqClosing 4; R</td><td>SeqClosing 8; O</td><td>SeqClosing 9; M</td></tr>
<tr><td>SeqClosing 10; R</td><td></td><td></td></tr>
<tr><td colspan="2">Test purpose</td><td colspan="3">Check that:

When an Endpoint generates a message that carries an RM protocol element in the body of a SOAP envelope that Endpoint MUST include in that envelope a wsa:Action SOAP header block whose value is an IRI that is a concatenation of the WS-RM namespace URI, followed by a "/", followed by the value of the local name of the child element of the SOAP body.

[AND]

To ensure that the Sequence ends with a known final state either the RM Source or RM Destination MAY choose to close the Sequence before terminating it.

[AND]

If the RM Source wishes to close the Sequence, then it sends a CloseSequence element, in the body of a message, to the RM Destination. This message indicates that the RM Destination MUST NOT accept any new messages for the specified Sequence, other than those already accepted at the time the CloseSequence element is interpreted by the RM Destination

[AND]

To allow the RM Destination to determine if it has received all of the messages in a Sequence, the RM Source SHOULD include the LastMsgNumber element in any CloseSequence messages it sends. The value of the LastMsgNumber element MUST be the same in all the CloseSequence messages for the closing Sequence

[AND]

The RM Source or RM Destination MUST include wsrm:Identifier element in any CloseSequence messages it sends. The RM Source or RM Destination MUST set the value of this element to the absolute URI (conformant with RFC3986) of the closing Sequence

[AND]

The RM Source SHOULD include wsrm:LastMessageNumber element in any CloseSequence message it sends.</td></tr>
<tr><td colspan="2">Applicability</td><td colspan="3">C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_WSI_032 AND C_SEN_GEN_003</td></tr>
<tr><td colspan="2">Other PICS</td><td colspan="3"></td></tr>
<tr><td colspan="2">Initial condition</td><td colspan="3">The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.</td></tr>
<tr><td colspan="2">Test procedure</td><td colspan="3">1. The HFS sender under test starts to send a Sequence message including an AckRequested element or indicating that it is the last message in the header block of the last message.
2. The simulated HFS receiver accepts all messages and if an offer was sent by the HFS sender, it also sends a Sequence message indicating that it is the last message.
3. The HFS sender sends with a SequenceAcknoledgement message.
4. If the HFS sender sends a CloseSequenceMessage then check the received message:
a. In the header block:
❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/CloseSequence.
b. In the body of the message, within the CloseSequence element:
❑ wsrm:Identifier value = an absolute URI of the closing sequence.
❑ The presence of wsrm:LastMsgNumber is recommended, and if it is present it must be the same in all CloseSequence elements of that closing sequence.
5. Or else, if C_SEN_WSI_032 = TRUE then force the HFS sender to close the sequence and check the received message.
a. In the header block:
❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/CloseSequence.</td></tr>
</table>

	b. In the body of the message, within the CloseSequence element: ❑ wsrm:Identifier value = an absolute URI of the closing sequence. ❑ The presence of wsrm:LastMsgNumber is recommended, and if it is present it must be the same in all CloseSequence elements of that closing sequence. 6. The simulated HFS receiver responds with a CloseSequenceResponse.
Pass/Fail criteria	All elements are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-005_B		
TP label		Closing a Sequence Response		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	WSAddress 1; C	SeqClosing 1; O	SeqClosing 11; M
		SeqClosing 12; M		
Test purpose		Check that: When an Endpoint generates a message that carries an RM protocol element in the body of a SOAP envelope that Endpoint MUST include in that envelope a wsa:Action SOAP header block whose value is an IRI that is a concatenation of the WS-RM namespace URI, followed by a "/", followed by the value of the local name of the child element of the SOAP body. [AND] To ensure that the Sequence ends with a known final state either the RM Source or RM Destination MAY choose to close the Sequence before terminating it. [AND] A wsrm:CloseSequenceResponse element is sent in the body of a message in response to receipt of a CloseSequence request message. It indicates that the responder has closed the Sequence [AND] The responder (RM Source or RM Destination) MUST include wsrm:Identifier element in any CloseSequenceResponse messages it sends. The responder MUST set the value of this element to the absolute URI (conformant with RFC3986) of the closing Sequence.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND NOT(C_SEN_WSI_032) AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.		
Test procedure		1. Run the HFS sender under test (make sure that the HFS sender has something, a measure or anything else, to send). 2. Wait until the HFS sender sends a CreateSequence message. 3. The simulated HFS receiver responds with a CreateSequenceResponse. If an offer is sent by the HFS sender in step 2, the HFS receiver accepts the offer. 4. The HFS sender under test starts to send a Sequence message including an AckRequested element or indicating that it is the last message in the header block of the last message. 5. The simulated HFS receiver accepts all messages and if an offer was sent by the HFS sender, it also sends a Sequence message indicating that it is the last message. 6. The HFS sender sends with a SequenceAcknoledgement message. 7. The simulated HFS receiver sends a CloseSequence element in the body of the message, including a correct LastMessageNumber. 8. The HFS sender responds with a CloseSequenceResponse message including: a. In the header block: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/CloseSequenceResponse.		

	b. In the body of the message: ❑ a CloseSequenceResponse element with a wsrm:Identifier element that is an absolute URI of the closing sequence response.
Pass/Fail criteria	All elements are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-006		
TP label		Sequence Termination		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	WSAddress 1; M	SeqTermination 1; R	SeqTermination 2; M
		SeqTermination 4; O	SeqTermination 5; M	SeqTermination 7; M
		SeqTermination 11; M	SeqTermination 12; R	
Test purpose		Check that: When an Endpoint generates a message that carries an RM protocol element in the body of a SOAP envelope that Endpoint MUST include in that envelope a wsa:Action SOAP header block whose value is an IRI that is a concatenation of the WS-RM namespace URI, followed by a "/", followed by the value of the local name of the child element of the SOAP body. [AND] To allow the RM Destination to determine if it has received all of the messages in a Sequence, the RM Source SHOULD include the LastMsgNumber element in any TerminateSequence messages it sends [AND] The value of the LastMsgNumber element in the TerminateSequence message MUST be equal to the value of the LastMsgNumber element in any CloseSequence message(s) sent by the RM Source for the same Sequence [AND] A wsrm:TerminateSequence element MAY be sent by an RM Source to indicate it has completed its use of the Sequence [AND] The RM Source MUST NOT send wsrm:TerminateSequence element as a header block [AND] Once wsrm:TerminateSequence element is sent, other than this element, the RM Source MUST NOT send any additional message to the RM Destination referencing this Sequence [AND] The RM Source or RM Destination MUST include wsrm:Identifier element in any TerminateSequence message it sends. The RM Source or RM Destination MUST set the value of this element to the absolute URI (conformant with RFC3986) of the terminating Sequence [AND] /wsrm:TerminateSequence/wsrm:LastMsgNumber. The RM Source SHOULD include this element in any TerminateSequence message it sends. The LastMsgNumber element specifies the highest assigned message number of all the Sequence Traffic Messages for the terminating Sequence.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_WSI_033 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.		
Test procedure		1. The HFS sender sends Sequence messages including an AckRequested element or indicating that it is the last message in the header block of the last message. 2. The HFS receiver under test responds using a SequenceAcknowledgement header block, accepting all messages.		

	3. If the HFS sender under test sends a TerminateSequence element in the body of the message, the expected messages are: a. In the header block: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/TerminateSequence ❑ wsrm: TerminateSequence is not present. b. In the body of the message, within the TerminateSequence element: ❑ wsrm:Identifier value is an absolute URI of the terminating sequence. ❑ It is recommended that a LastMsgNumber element is present, and, if present, it must be equal to the LastMsgNumber of any CloseSequence message. 4. If the HFS sender has sent a TerminateSequence element, the simulated HFS receiver responds with a TerminateSequenceResponse message, including its Identifier element as an absolute URI. 5. Once the sequence is terminated, the HFS sender under test does not send any message referencing that terminated sequence.
Pass/Fail criteria	All elements are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-006_B		
TP label		Sequence Termination Response		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	WSAddress 1; M	SeqTermination 10; M	SeqTermination 13; M
		SeqTermination 14; M	SeqTermination 15; M	
Test purpose		Check that: When an Endpoint generates a message that carries an RM protocol element in the body of a SOAP envelope that Endpoint MUST include in that envelope a wsa:Action SOAP header block whose value is an IRI that is a concatenation of the WS-RM namespace URI, followed by a "/", followed by the value of the local name of the child element of the SOAP body. [AND] Upon receipt of a TerminateSequence the RM Source MUST NOT send any additional messages (with the exception of the corresponding TerminateSequenceResponse) for this Sequence. [AND] TeminateSequenceResponse element is sent in the body of a message in response to receipt of a TerminateSequence request message. It indicates that the responder has terminated the Sequence. The responder MUST NOT send this element as a header block [AND] The responder (RM Source or RM Destination) MUST include this element in any TerminateSequenceResponse message it sends. The responder MUST set the value of this element to the absolute URI (conformant with RFC3986) of the terminating Sequence. [AND] On receipt of a TerminateSequence message the HFS receiver (RM Source or RM Destination) MUST respond with a corresponding TerminateSequenceResponse message or generate a fault UnknownSequenceFault if the Sequence is not known.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND NOT(C_SEN_WSI_033) AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.		
Test procedure		1. The HFS sender sends Sequence messages including an AckRequested element or indicating that it is the last message in the header block of the last message. 2. The HFS receiver under test responds using a SequenceAcknowledgement header		

	block, accepting all messages. 3. The simulated HFS receiver sends a TerminateSequence element in the body of the message, with a correct LastMsgNumber. 4. The HFS sender responds only with a message including: a. In the header block: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/TerminateSequenceResponse ❑ wsrm: TerminateSequenceResponse is not present. b. In the body of the message within the TerminateSequenceResponse element: ❑ wsrm:Identifier element as an absolute URI of the terminating sequence. 5. Once the sequence is terminated, the HFS sender under test does not send any message referencing that terminated sequence.
Pass/Fail criteria	All elements are as specified within the test procedure above.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-007		
TP label		Sequences		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	ProtocolInv 1; M	Sequences 1; M	Sequences 2; M
		Sequences 3; M	Sequences 5; M	Sequences 6; M
		Sequences 7; M	Sequences 8; M	
Test purpose		Check that: The RM Source MUST assign each message within a Sequence a message number beginning at 1 and increasing by exactly 1 for each subsequent message. These numbers MUST be assigned in the same order in which messages are sent by the Application Source. [AND] The RM Source MUST include a Sequence header block in all messages for which reliable transfer is REQUIRED [AND] The RM Source MUST identify Sequences with unique Identifier elements and the RM Source MUST assign each message within a Sequence a MessageNumber element that increments by 1 from an initial value of 1 [AND] The RM Source MUST NOT include more than one Sequence header block in any message [AND] The RM Source MUST assign a mustUnderstand attribute with a value 1/true (from the namespace corresponding to the version of SOAP to which the Sequence SOAP header block is bound) to the Sequence header block element. [AND] An RM Source that includes a Sequence header block in a SOAP envelope MUST include wsrm:Identifier element in that header block [AND] The RM Source MUST set the value of wsrm:Identifier element to the absolute URI (conformant with RFC3986) that uniquely identifies the Sequence [AND] The RM Source MUST include wsrm:MessageNumber element within any Sequence headers it creates. This element is of type MessageNumberType.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				

Initial condition	The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.
Test procedure	1. Wait until the HFS sender under test sends Sequence message/s including an AckRequested element or indicating that it is the last message in the last message header block. 2. The expected message/s are: ❑ wsrm:MessageNumber element is of type MessageNumberType and starts in 1 and increments by 1 in every sequential message. ❑ There is only one Sequence header block in each message. ❑ wsrm:Identifier element must be present in the header block and must be an absolute URI that uniquely identifies the sequence. ❑ mustUnderstand attribute = "1" or "true". 3. The simulated HFS receiver responds using a SequenceAcknowledgement header block accepting all messages received.
Pass/Fail criteria	All elements are as specified in step 2.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-010		
TP label		Unknown Sequence Fault		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	UnknownSeq 1; M	UnknownSeq 2; M	UnknownSeq 3; M
		Faults 1; R	Faults 2; M	Faults 3; M
Test purpose		Check that: UnknownSequence has the following properties: [Code] HFS Sender [Subcode] wsrm:UnknownSequence [Reason] The value if wsrm:Identifier is not a known Sequence identifier [Detail] <wsrm:Identifier ...> xs:anyURI </wsrm:Identifier> [AND] An Endpoint MUST generates an UnknownSequence fault in response to a message containing an unknown or terminated Sequence identifier [AND] An Endpoint that receives an UnknownSequence fault MUST terminate the Sequence if not otherwise terminated [AND] Destinations that generate faults related to known sequences SHOULD transmit those faults. [AND] If transmitted, faults MUST be transmitted to the same [destination] as Acknowledgement messages [AND] Entities that generate WS-ReliableMessaging faults MUST include as the [action] property the default fault action IRI: http://docs.oasis-open.org/ws-rx/wsrm/200702/fault.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_WSI_034 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "None" sequence state. The simulated HFS receiver is able to send a CloseSequence message in the "None" sequence state.		
Test procedure		1. The simulated HFS receiver transmits a CloseSequence message with an unknown		

	identifier. 2. The HFS sender under test generates an UnknownSequence fault. It is recommended that the fault is transmitted to the HFS receiver. 3. That message includes the following properties: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/fault ❑ Code = HFS Sender ❑ Subcode = wsrm:UnknownSequence ❑ Reason = The value if wsrm:Identifier is not a known Sequence identifier ❑ Detail = <wsrm:Identifier...> xs:anyURI </wsrm:Identifier>.
Pass/Fail criteria	All elements are as specified in step 3.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-011		
TP label		Invalid Acknowledgement Fault		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	InvalidAck 1; M	InvalidAck 2; M	Faults 1; R
		Faults 2; M	Faults 3; M	
Test purpose		Check that: InvalidAcknowledgement fault has the following properties: [Code] HFS Sender [Subcode] wsrm:InvalidAcknowledgement [Reason] The SequenceAcknowledgement violates the cumulative Acknowledgement invariant. [Detail] <wsrm:SequenceAcknowledgement ...> ... </wsrm:SequenceAcknowledgement> [AND] RM Source MUST generate an InvalidAcknowledgement in response to a SequenceAcknowledgement that violate the invariants stated in 2.3 or any of the requirements in 3.9 about valid combinations of AckRange, Nack and None in a single SequenceAcknowledgement element or with respect to already Received such elements. [AND] Destinations that generate faults related to known sequences SHOULD transmit those faults. [AND] If transmitted, faults MUST be transmitted to the same [destination] as Acknowledgement messages [AND] Entities that generate WS-ReliableMessaging faults MUST include as the [action] property the default fault action IRI: http://docs.oasis-open.org/ws-rx/wsrm/200702/fault.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_WSI_034 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state.		
Test procedure		1. The HFS sender under test starts to send Sequence messages with their respective message number. 2. Wait until the HFS sender sends an AckRequested element or indicates that the message is the last one. 3. The simulated HFS receiver responds with a SequenceAcknowledgement with an AckRange, a None and a Nack element. 4. The HFS sender generates an InvalidAcknowledgement fault. It is recommended that the		

	fault is transmitted to the HFS receiver. 5. That message includes the following properties: ❑ wsa:Action = http://docs.oasis-open.org/ws-rx/wsrm/200702/fault ❑ Code = HFS Sender ❑ Subcode = wsrm:InvalidAcknowledgement ❑ Reason = <any> ❑ Detail = <any related to the message that produces the fault>.
Pass/Fail criteria	All elements are as specified in step 5.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-012		
TP label		Message Number Rollover		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	MessageNumrRoll 4; R		
Test purpose		Check that: RM Source SHOULD continue to retransmit undelivered messages until the Sequence is closed or terminated.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "Created" sequence state. The simulated HFS receiver is able to send a MessageNumberRollover fault instead of a SequenceAcknowledgement message.		
Test procedure		1. The HFS sender under test transmits a Sequence message. 2. The simulated HFS receiver generates a MessageNumberRollover fault, which is transmitted to the HFS sender. 3. The HFS sender should retransmit undelivered messages until the HFS receiver closes or terminates the sequence.		
Pass/Fail criteria		The HFS sender should retransmit undelivered messages in step 3.		
Notes				

TP Id		TP/HFS/SEN/WSI/RM/BV-012_A		
TP label		Create Sequence Refused		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	SeqRefused 3; M		
Test purpose		Check that: The Action Upon Reception is Sequence Terminated when the HFS receiver does not wish to create a new Sequence.		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "None" sequence state. The simulated HFS receiver is able to send a CreateSequenceRefused fault instead of a CreateSequenceResponse message.		
Test procedure		1. Wait until the HFS sender under test sends a CreateSequence message to the simulated HFS receiver. 2. The simulated HFS receiver responds with a CreateSequenceRefused fault. 3. The HFS sender must terminate the sequence.		

Pass/Fail criteria	The HFS sender terminates the sequence when it receives a CreateSequenceRefused fault.
Notes	

TP Id		TP/HFS/SEN/WSI/RM/BV-012_B		
TP label		Sequence Closed Fault		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	SeqClosedFault 3; M		
Test purpose		Check that: The Action Upon Reception is Sequence Closed		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the CreatedSequence state. The simulated HFS receiver is able to send a SequenceClosed fault instead of a SequenceAcknowledgement message.		
Test procedure		1. The HFS sender under test sends a sequence to the simulated HFS receiver sending an AckRequested message or indicating that it is the last message. 2. The simulated HFS receiver sends a SequenceClosed fault. 3. The HFS sender must close the sequence.		
Pass/Fail criteria		The HFS sender closes the sequence when it receives a SequenceClosed fault.		
Notes				

TP Id		TP/HFS/SEN/WSI/RM/BV-015		
TP label		Securing Sequences Using WS-Security		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	SecSeqWSS 5; R	SecSeqWSS 6; R	
Test purpose		Check that: The RM Source SHOULD include the UsesSequenceSTR element as a SOAP header block within the CreateSequence message. This element MUST include a soap:mustUnderstand attribute with a value of "true".		
Applicability		C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_WSI_003 AND C_SEN_GEN_003		
Other PICS				
Initial condition		The HFS sender under test and the simulated HFS receiver are in the "None" sequence state.		
Test procedure		1. Wait until the HFS sender under test sends a CreateSequence message. 2. It is recommended that the received message includes a UsesSequenceSTR element in the header block. If the element is included, it MUST include a soap:mustUnderstand attribute = "true".		
Pass/Fail criteria		The recommended element in step 2 is as specified within the test procedure above.		
Notes				

TP Id		TP/HFS/SEN/WSI/RM/BV-016		
TP label		Securing Sequences Using SSL/TLS		
Coverage	**Spec**	[OASIS WS-I RM]		
	Testable items	SecSeqSSL/TLS 1; M	SecSeqSSL/TLS 2; O	SecSeqSSL/TLS 3; M

Test purpose	Check that: If the RM Source wishes to bind a Sequence to the underlying SSL/TLS sessions(s) it MUST include the UsesSequenceSSL element as a SOAP header block within the CreateSequence message. [AND] The RM Source MAY include wsrm:UsesSequenceSSL element as a SOAP header block of a CreateSequence message to indicate to the RM Destination that the resulting Sequence is to be bound to the TLS session that was used to carry the CreateSequence message [AND] If wsrm:UsesSequenceSSL element is included, the RM Source MUST mark this header with a soap:mustUnderstand attribute with a value of "true".
Applicability	C_SEN_000 AND C_SEN_WSI_021 AND C_SEN_GEN_003
Other PICS	
Initial condition	The HFS sender under test and the simulated HFS receiver are in the "None" sequence state.
Test procedure	1. Wait until the HFS sender under test sends a CreateSequence message. 2. If the HFS sender binds a sequence to the underlying SSL/TLS sessions(s) it includes the UsesSequenceSSL element as a SOAP header block within the CreateSequence message, with a soap:mustUnderstand attribute = "true".
Pass/Fail criteria	If the HFS sender binds the sequence to the underlying TSL session, elements are as specified in step 2.
Notes	

Bibliography

[b-ITU-T H.810 (2013)] Recommendation ITU-T H.810 (2013), *Interoperability design guidelines for personal health systems*.

[b-ITU-T H.810 (2015)] Recommendation ITU-T H.810 (2015), *Interoperability design guidelines for personal health systems*.

[b-CDG 1.0] Continua Health Alliance, Continua Design Guidelines v1.0 (2008), *Continua Design Guidelines.*

[b-CDG 2010] Continua Health Alliance, Continua Design Guidelines v1.5 (2010), *Continua Design Guidelines.*

[b-CDG 2011] Continua Health Alliance, Continua Design Guidelines (2011), "Adrenaline", *Continua Design Guidelines.*

[b-CDG 2012] Continua Health Alliance, Continua Design Guidelines (2012), "Catalyst", *Continua Design Guidelines.*

[b-CDG 2013] Continua Health Alliance, Continua Design Guidelines (2013), "Endorphin" *Continua Design Guidelines.*

[b-CDG 2015] Continua Health Alliance, Continua Design Guidelines (2015), "Genome", *Continua Design Guidelines*.

[b-CDG 2016] Personal Connected Health Alliance, Continua Design Guidelines (2016), "Iris", *Continua Design Guidelines*.

[b-ETSI SR 001 262] ETSI SR 001 262 v1.8.1 (2003), *ETSI drafting rules*.
https://docbox.etsi.org/MTS/MTS/10-PromotionalMaterial/MBS-20111118/Referenced%20Documents/Drafting%20Rules.pdf

[b-HFSR PICS & PIXIT] Services HFS Receiver DG2016 PICS and PIXIT excel sheet v1.7
http://handle.itu.int/11.1002/2000/12067

[b-HFSS PICS & PIXIT] Services HFS Sender DG2016 PICS and PIXIT excel sheet v1.7
http://handle.itu.int/11.1002/2000/12067

[b-SOAP 1.2] W3C SOAP 1.2 (2007), *SOAP Version 1.2 (Second Edition)*.
http://www.w3.org/TR/soap

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2017