



Enabler Test Specification for Presence SIMPLE

Interoperability

Candidate Version 1.1 – 31 Mar 2007

Open Mobile Alliance
OMA-ETS-Presence_SIMPLE_INT-V1_1-20080331-C

Use of this document is subject to all of the terms and conditions of the Use Agreement located at <http://www.openmobilealliance.org/UseAgreement.html>.

Unless this document is clearly designated as an approved specification, this document is a work in process, is not an approved Open Mobile Alliance™ specification, and is subject to revision or removal without notice.

You may use this document or any part of the document for internal or educational purposes only, provided you do not modify, edit or take out of context the information in this document in any manner. Information contained in this document may be used, at your sole risk, for any purposes. You may not use this document in any other manner without the prior written permission of the Open Mobile Alliance. The Open Mobile Alliance authorizes you to copy this document, provided that you retain all copyright and other proprietary notices contained in the original materials on any copies of the materials and that you comply strictly with these terms. This copyright permission does not constitute an endorsement of the products or services. The Open Mobile Alliance assumes no responsibility for errors or omissions in this document.

Each Open Mobile Alliance member has agreed to use reasonable endeavors to inform the Open Mobile Alliance in a timely manner of Essential IPR as it becomes aware that the Essential IPR is related to the prepared or published specification. However, the members do not have an obligation to conduct IPR searches. The declared Essential IPR is publicly available to members and non-members of the Open Mobile Alliance and may be found on the “OMA IPR Declarations” list at <http://www.openmobilealliance.org/ipr.html>. The Open Mobile Alliance has not conducted an independent IPR review of this document and the information contained herein, and makes no representations or warranties regarding third party IPR, including without limitation patents, copyrights or trade secret rights. This document may contain inventions for which you must obtain licenses from third parties before making, using or selling the inventions. Defined terms above are set forth in the schedule to the Open Mobile Alliance Application Form.

NO REPRESENTATIONS OR WARRANTIES (WHETHER EXPRESS OR IMPLIED) ARE MADE BY THE OPEN MOBILE ALLIANCE OR ANY OPEN MOBILE ALLIANCE MEMBER OR ITS AFFILIATES REGARDING ANY OF THE IPR'S REPRESENTED ON THE “OMA IPR DECLARATIONS” LIST, INCLUDING, BUT NOT LIMITED TO THE ACCURACY, COMPLETENESS, VALIDITY OR RELEVANCE OF THE INFORMATION OR WHETHER OR NOT SUCH RIGHTS ARE ESSENTIAL OR NON-ESSENTIAL.

THE OPEN MOBILE ALLIANCE IS NOT LIABLE FOR AND HEREBY DISCLAIMS ANY DIRECT, INDIRECT, PUNITIVE, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR EXEMPLARY DAMAGES ARISING OUT OF OR IN CONNECTION WITH THE USE OF DOCUMENTS AND THE INFORMATION CONTAINED IN THE DOCUMENTS.

© 2008 Open Mobile Alliance Ltd. All Rights Reserved.

Used with the permission of the Open Mobile Alliance Ltd. under the terms set forth above.

Contents

1. SCOPE	5
2. REFERENCES	6
2.1 NORMATIVE REFERENCES	6
2.2 INFORMATIVE REFERENCES	6
3. TERMINOLOGY AND CONVENTIONS	7
3.1 CONVENTIONS	7
3.2 DEFINITIONS	7
3.3 ABBREVIATIONS	7
3.4 TESTING POLICIES	8
3.5 TESTING ASSUMPTIONS	8
4. INTRODUCTION	9
5. PRESENCE-SIMPLE INTEROPERABILITY TEST CASES	10
5.1 PRESENCE FEATURES	10
5.1.1 Normal Flow	10
5.1.1.1 <i>Presence-1.1-int-0100 Publication of Presence information</i>	10
5.1.1.2 <i>Presence-1.1-int-0101 Publication of Presence information, publish modification</i>	11
5.1.1.3 <i>Presence-1.1-int-0102 Publication of Presence information, removal</i>	12
5.1.1.4 <i>Presence-1.1-int-0104 Publication of Presence information, subscription removal</i>	13
5.1.1.5 <i>Presence-1.1-int-0105 Publication of Presence information, subscription refresh</i>	14
5.1.1.6 <i>Presence-1.1-int-0106 Partial Publication of Presence information (Includes Optional Features)</i>	15
5.1.1.7 <i>Presence-1.1-int-0107 Notification of Presence information from multiple Presentities</i>	15
5.1.1.8 <i>Presence-1.1-int-0108 Partial Notification of Presence information (Includes Optional Features)</i>	16
5.1.1.9 <i>Presence-1.1-int-0109 Distribution Policy (Presence Content Rules I)</i>	18
5.1.1.10 <i>Presence-1.1-int-0110 Distribution Policy (Presence Content Rules II)</i>	19
5.1.1.11 <i>Presence-1.1-int-0111 Combining presence elements from different presence sources</i>	21
5.1.1.12 <i>Presence-1.1-int-0120 Publication of presence information, Subscription Poll Request</i>	22
5.1.1.13 <i>Presence-1.1-int-0121 Anonymous Distribution Policy</i>	23
5.1.1.14 <i>Presence-1.1-int-0122 Default Policy</i>	24
5.1.1.15 <i>Presence-1.1-int-0123 Authorization management for groups</i>	24
5.1.1.16 <i>Presence-1.1-int-0124 Combining permissions on an ongoing subscription</i>	26
5.1.1.17 <i>Presence-1.1-int-0130 Publication of presence information, Watcher is blocked</i>	28
5.1.1.18 <i>Presence-1.1-int-0140 Publication of presence information, Watcher is politely blocked</i>	29
5.1.1.19 <i>Presence-1.1-int-0141 Notification Filtering (Includes Optional Features)</i>	30
5.1.1.20 <i>Presence-1.1-int-0142 Event Notification Filtering (Includes Optional Features)</i>	31
5.1.1.21 <i>Presence-1.1-int-0143 Notification of Only Authorized Presence Information is sent to Watchers (Includes Optional Features)</i>	32
5.1.1.22 <i>Presence-1.1-int-0144 Reactive Authorization for a specific group (Includes Optional Features)</i>	34
5.1.2 Error Flow	35
5.1.2.1 <i>Presence-1.1-int-0145 Publication of Presence information not supported by watcher</i>	35
5.2 RESOURCE LIST TEST CASES	36
5.2.1 Normal Flow	36
5.2.1.1 <i>Presence-1.1-int-0150 Subscription to a resource list</i>	36
5.2.1.2 <i>Presence-1.1-int-0151 Adding a Presentity to an ongoing list subscription</i>	37
5.2.1.3 <i>Presence-1.1-int-0152 Subscription to shared lists</i>	38
5.2.1.4 <i>Presence-1.1-int-0153 Adding a Presentity to an ongoing shared list subscription</i>	39
5.2.1.5 <i>Presence-1.1-int-0160 RLS Event notification filtering (Includes Optional Features)</i>	40
5.2.2 Error Flow	41
5.3 WATCHER INFORMATION SUBSCRIBER TEST CASES	42
5.3.1 Normal Flow	42
5.3.1.1 <i>Presence-1.1-int-0200 Subscribe to Watcher Information</i>	42
5.3.1.2 <i>Presence-1.1-int-0210 Publication of presence information, Watcher is pending</i>	43
5.3.1.3 <i>Presence-1.1-int-0211 Notification to Watcher Information Subscriber when a subscription to His/Her Presence Information Expires (Includes Optional features)</i>	44
5.3.2 Error Flow	45
APPENDIX A. CHANGE HISTORY (INFORMATIVE)	46

A.1	APPROVED VERSION HISTORY	46
A.2	DRAFT/CANDIDATE VERSION 1.1 HISTORY	46
APPENDIX B.	SCR AND SPECIFICATION REFERENCES.....	47

1. Scope

This document describes in detail available test cases for Presence SIMPLE V1.1 enabler (<http://www.openmobilealliance.org>).

This test specification only covers interoperability test cases.

The interoperability test cases are aimed to verify that implementations of the specifications work satisfactory.

If tests do not exist at the creation of the test specification this part should be marked not available.

2. References

2.1 Normative References

- [ERELD] “Enabler Release Document for Presence”, Open Mobile Alliance™, OMA-ERELD-SIMPLE-V1_1, URL:<http://www.openmobilealliance.org/>
- [IOPPROC] “OMA Interoperability Policy and Process”, Version 1.5, Open Mobile Alliance™, OMA-IOP-Process-V1_5, URL:<http://www.openmobilealliance.org/>
- [OMA-Presence-XDM] “Presence XDM Specification”, Version 1.1, Open Mobile Alliance™, OMA-Presence_SIMPLE_XDM_Specification-V1_1, URL:<http://www.openmobilealliance.org/>
- [OMARDPOC] “Push to Talk over Cellular Requirements”, Version 1.0, Open Mobile Alliance™, OMA-RD_PoC-V1_0, URL:<http://www.openmobilealliance.org/>
- [OMARDPRES] “Presence Requirements”, Version 1.1, Open Mobile Alliance™, OMA-RD_Presence_SIMPLE-V1_1, URL:<http://www.openmobilealliance.org/>
- [OMA-RLS-XDM] “Resource List Server (RLS) XDM Specification”, Version 1.1, Open Mobile Alliance™, OMA-Presence_SIMPLE_RLS_XDM_Specification-V1_1, URL:<http://www.openmobilealliance.org/>
- [OMATSPRES] “Presence SIMPLE Specification”, Version 1.1, Open Mobile Alliance™, OMA-TS-Presence_SIMPLE-V1_1, URL:<http://www.openmobilealliance.org/>
- [RFC2119] “Key words for use in RFCs to Indicate Requirement Levels”, S. Bradner, March 1997, URL:<http://www.ietf.org/rfc/rfc2119.txt>

2.2 Informative References

- [OMAADPRES] “Stage 2 - Presence using SIMPLE”, Version 1.1, Open Mobile Alliance™, OMA-AD-Presence_SIMPLE-V1_1, URL:<http://www.openmobilealliance.org/>
- [OMADICT] “Dictionary for OMA Specifications”, Open Mobile Alliance™. OMA-Dictionary, URL:<http://www.openmobilealliance.org/>

3. Terminology and Conventions

3.1 Conventions

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope”, are normative, unless they are explicitly indicated to be informative.

Following test case numbering scheme is followed in the ETS for different Test Sections.

Note: In following numbering scheme “int” stands for “Interoperability Test Cases”. E.g. the first Interoperability test cases will be “Presence-1.1-int-0100”.

Following is the definition of fields in the naming convention:

PRES-1.1	Int	01	00
Specification Release (Presence-SIMPLE Version) number.	Int – Interoperability	Test-category	Test Sequence Number

3.2 Definitions

UE[N]	A client terminal with assigned User[N], used for testing where N is an integer number (i.e. UE1, UE2, etc.).
user	A person using UE.
User[N]	A subscriber assigned to UE, where N is an integer number (i.e. User1, User2, etc.)

3.3 Abbreviations

OMA	Open Mobile Alliance
PoC	Push to talk over Cellular
PS	Presence Server
RD	Requirements Document
RLS	Resource List Server
SIP	Session Initiation Protocol
URI	Universal Resource Identifier
XCAP	XML Configuration Access Protocol
XDMC	XML Document Management Client
XDMS	XML Document Management Server
XML	Extensible Mark-up Language

3.4 Testing Policies

This section is intended to describe the testing policies used throughout the document.

For the UE testing, the focus is on UE1 (from Company1), UE2 (from Company2) and UE3 (from either Company). UE1, UE2, RLS and the Presence Server MUST be from different vendors.

There are no vendor restrictions for UE3 and XDMS.

3.5 Testing Assumptions

For all test cases throughout the document, the following assumptions are valid unless stated otherwise. Therefore, these assumptions shall be seen as a part of the preconditions:

General:

- The UE will contain a Presence Source and/or a Watcher, and is able to communicate with a Presence Server, Resource List Server, and Shared, Presence and RLS XDM Servers.
- The UE is able to show presence information.

4. Introduction

The purpose of this document is to provide test cases for Presence SIMPLE Enabler Release V1.1.

The following items on an overall level are needed to adequately test the Presence SIMPLE Enabler:

- Clients that contains Watcher and Presence Source logical components
- Clients capable of manipulating its Authorization Rules Document that is stored in Presence XDMS
- Presence Server
- Shared, Presence and RLS XDMS with Aggregation Proxy
- Resource List Server

Detailed information will be included in the specific test case descriptions.

The Presence SIMPLE Enabler tests are carried out using XCAP and SIP protocols. The transport protocols used are UDP, TCP and TLS.

5. Presence-SIMPLE Interoperability Test Cases

- This section covers the test cases defined for interoperability testing of the Presence SIMPLE enabler.

5.1 Presence Features

This section lists the test cases that can be executed by the following configurations of implementations:

- Client with XDMC, Presence Source and Watcher functionality.
- Aggregation Proxy with a Presence XDMS

5.1.1 Normal Flow

5.1.1.1 Presence-1.1-int-0100 Publication of Presence information

Test Case Id	Presence-1.1-int-0100
Test Object	UE with Presence Source and UE with Presence Watcher functionality, Presence Server.
Test Case Description	Verify that presence information published by an UE will be received by another UE, which subscribes for that information. <u>TEST CASE GOAL:</u> Verify that when UE1 publishes presence information, UE2, as Watcher, will receive the presence information.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications ○ User2 has no active subscriptions for User1
Test Procedure	<ol style="list-style-type: none"> 1. User2 subscribes to presence information from User1. 2. User1 publishes presence information for all commonly supported Presence elements.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information published by User1

5.1.1.2 Presence-1.1-int-0101 Publication of Presence information, publish modification

Test Case Id	Presence-1.1-int-0101
Test Object	UEs with Presence Source and Presence Watcher functionality, Presence Server.
Test Case Description	Verify that presence information modified by an UE will be displayed accordingly in another UE, which subscribes for that information. <u>TEST CASE GOAL:</u> Verify that when User1 modifies presence information, User2, as Watcher, will receive the updated presence information.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has an active subscription to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User1 modifies the presence information that has already been published, e.g. change of mood.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the updated presence information related to User1.

5.1.1.3 Presence-1.1-int-0102 Publication of Presence information, removal

Test Case Id	Presence-1.1-int-0102
Test Object	UEs with Presence Source and Presence Watcher functionality, Presence Server.
Test Case Description	<p>Verify that presence publications terminated by an UE will be displayed in another UE.</p> <p><u>TEST CASE GOAL:</u> Verify that when User1 terminates its presence publication, User2, as Watcher, will be displayed.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE1 capable to terminate its publication (expiration Header = 0) ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has an active publication. ○ User2 has an active subscription to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 modifies the active presence information that has already been published. 2. User1 removes (terminates) his publication.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE2 displays the presence information published by User1. 2. UE2 displays that User1 is not available.

5.1.1.4 Presence-1.1-int-0104 Publication of Presence information, subscription removal

Test Case Id	Presence-1.1-int-0104
Test Object	UEs with Presence Source and presence Watcher functionality, Presence Server
Test Case Description	An UE, acting as a Watcher terminates its subscriptions, and another UE, the presence source, updates the presence information. <u>TEST CASE GOAL:</u> Verify that a Watcher, which has terminated its subscription, does not display any presence updates.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no subscription to User1 active. ○ UE2 capable of removing its subscription. (Expires Header = 0)
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information from User1. 3. User2 removes the subscription to presence information from User1. 4. User1 modifies the active presence information that has already been published.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information published by User1. 4. UE2 does not display any presence information from User1.

5.1.1.5 Presence-1.1-int-0105 Publication of Presence information, subscription refresh

Test Case Id	Presence-1.1-int-0105
Test Object	UEs with Presence Source and presence Watcher functionality, Presence Server
Test Case Description	<p>Verify that Presence Server keeps sending presence information to a UE, acting as a watcher, after subscription refresh.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE retrieves and displays the presence information after the subscription refresh.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information. ○ UE2 is capable of setting expiration time of a subscription. ○ Presence Server permits a subscription expiration time of 60 seconds. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no subscription to User1 active.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information from User1 using a particular expiration time (e.g. expiration time = 60 seconds). 3. Within the time subscription duration (60 seconds), User2 subscribes again (subscription refresh) to presence information related to presence information of User1 (expiration time = 60 seconds). 4. User1 modifies its presence information.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information published by User1. 4. UE2 displays the updated presence information published by User1.

5.1.1.6 Presence-1.1-int-0106 Partial Publication of Presence information (Includes Optional Features)

Test Case Id	Presence-1.1-int-0106
Test Object	UEs with Presence Source with partial publication capabilities and Presence Watcher functionality, Presence Server with partial publication capabilities.
Test Case Description	<p>Verify that presence information modified by an UE via partial publication will be displayed accordingly in another UE, which subscribes for that information.</p> <p><u>TEST CASE GOAL:</u> Verify that when User1 modifies presence information via partial publication, User2, as Watcher, will receive the updated presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has an active subscription to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User1 modifies via partial publication the presence information that has already been published, e.g. change of mood.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the updated presence information related to User1.

5.1.1.7 Presence-1.1-int-0107 Notification of Presence information from multiple Presentities

Test Case Id	Presence-1.1-int-0107
Test Object	UEs with Presence Source and Presence Watcher functionality, Presence Server.
Test Case Description	Verify that a Presence Server can store and manage presence information coming from multiple UEs, acting as Presence Sources and related to several Users, and correctly notify one UE, acting as a Watcher the presence information.

	<u>TEST CASE GOAL:</u> Verify that a UE, acting as a Watcher, is able to display the presence information when subscribing to presence information of several other users.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active publications. ○ User3 has two active subscriptions; to User1 and User2.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 publishes presence information for all commonly supported Presence elements.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE3 displays the presence information published by User1. 2. UE3 displays the presence information published by User2.

5.1.1.8 Presence-1.1-int-0108 Partial Notification of Presence information (Includes Optional Features)

Test Case Id	Presence-1.1-int-0108
Test Object	UEs with Presence Watcher with partial notification capabilities and Presence Source functionality, Presence Server with partial notification capabilities.
Test Case Description	<p>Verify that presence information modified by an UE will be displayed accordingly in another UE, which subscribes via partial subscription to that information.</p> <p><u>TEST CASE GOAL:</u> Verify that when User1 modifies presence information, User2, as Watcher, will receive the updated presence information via partial notification.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A

Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active subscription to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User 2 subscribes via partial subscription to presence information from User 1. 2. User1 publishes presence information for all commonly supported Presence elements. 3. User1 modifies the presence information that has already been published, e.g. change of mood.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information related to User1 via partial notification. 3. UE2 displays the updated presence information related to User1 via partial notification.

5.1.1.9 Presence-1.1-int-0109 Distribution Policy (Presence Content Rules I)

Test Case Id	Presence-1.1-int-0109
Test Object	UEs with Presence Source, Presence Watcher and XDMC functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a User is able to define policies so that different presence information can be sent to different Users, acting as Watchers.</p> <p><u>TEST CASE GOAL:</u> Verify that a User, acting as a Presentity can allow one User to see a different presence content than another User, acting as Watchers.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains no information about User2 and User3 authorizations to see any of the presence information belonging to User1. ○ UE2, UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has an active publication with all commonly supported Presence elements. ○ User1 can change his authorizations rules document. ○ User2 and User3 have no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 sets authorizations rules and content access for User2. 2. User1 sets authorizations rules and content access for User3 (different than for User2). 3. User2 subscribes to presence information from User1. 4. User3 subscribes to presence information from User1
Pass-Criteria	<ol style="list-style-type: none"> 3. UE2 displays the presence information published by User1 according to the rules for User2 set by User1. 4. UE3 displays the presence information published by User1 according to the rules for User3 set by User1

5.1.1.10 Presence-1.1-int-0110 Distribution Policy (Presence Content Rules II)

Test Case Id	Presence-1.1-int-0110
Test Object	UEs with Presence Source, Presence Watcher and XDMC functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a User is able to define policies so that the same presence information elements but with different can be sent to different Users, acting as Watchers.</p> <p><u>TEST CASE GOAL:</u> Verify that a User, acting as a Presentity can allow one User to see the same presence information elements but with different values than another User, acting as Watchers.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains no information about User2 and User3 authorizations to see any of the presence information belonging to User1. ○ UE2, UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publication. ○ User1 can change his authorizations rules document. ○ User2 and User3 have no active subscriptions.

Test Procedure	<ol style="list-style-type: none"> 1. User1 sets authorizations rules and content access for User2, allowing User2 to see elements of his presence information (tuple/s, device/s and/or person/s) whose class subelement is “forUser2” 2. User1 sets authorizations rules and content access for User3, allowing User3 to see elements of his presence information (tuple/s, device/s and/or person/s) whose class subelement is “forUser3”. 3. User1 publishes presence information for all his supported Presence elements (tuple/s, device/s and/or person/s), some with class subelement “forUser2” and some with class subelement “forUser3”.. 4. User2 subscribes to presence information from User1. 5. User3 subscribes to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 4. UE2 receives and displays only the elements (tuple/s, device/s and/or person/s) with class subelement “forUser2” published by User1 5. UE3 receives and displays only the elements (tuple/s, device/s and/or person/s) with class subelement “forUser3” published by User1

5.1.1.11 Presence-1.1-int-0111 Combining presence elements from different presence sources

Test Case Id	Presence-1.1-int-0111
Test Object	UEs with Presence Source and Presence Watcher functionality, Presence Server.
Test Case Description	<p>Verify that a Presence Server supports the combination of different presence information elements of a particular User coming from different UEs, acting as presence sources.</p> <p><u>TEST CASE GOAL:</u> Verify that a Presence Server is able to apply presence composition rules, and notify a UE, acting as a Watcher, the correct presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N//A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (UE1 and UE2 with User1 credentials) ○ 1 UE (UE3 with User2 credential) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE3 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User2 has an active subscription to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 using UE1 publishes its own presence information related to User1. 2. User1 using UE2 publishes its own presence information related to User1 with different kind and/or values elements than UE1.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE3 displays the presence information published by User1, according to the presence composition rules.

5.1.1.12 Presence-1.1-int-0120 Publication of presence information, Subscription Poll Request

Test Case Id	Presence-1.1-int-0120
Test Object	UEs with presence source and presence Watcher functionality, Presence Server.
Test Case Description	Verify that a UE successfully publishes and retrieves presence information by polling. <u>TEST CASE GOAL:</u> Verify that one user using Polling Subscription, will retrieve presence information from another user, which has an active publication.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ User1 and User2 will have a set of commonly supported Presence elements ○ UE2 is capable of displaying presence information. ○ User1 has no active publications. ○ User2 has no active subscriptions. ○ UE2 has the capability to subscribe for presence information using poll.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information from User1 using poll (Expires Header = 0). 3. User1 modifies the active presence information that has been published.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information published by User1. 3. UE2 does not display the updated presence information by User1.

5.1.1.13 Presence-1.1-int-0121 Anonymous Distribution Policy

Test Case Id	Presence-1.1-int-0121
Test Object	UEs with Presence Source, Presence Watcher and XDMS functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a User is able to define policies so that defined presence information can be sent to an anonymous User, acting as a Watcher.</p> <p><u>TEST CASE GOAL:</u> Verify that a User, acting as a Presentity can define the contents a User authenticated as anonymous and acting as a Watcher will see.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document for User1 contains information about what presence information anonymous Users are allowed to see. User3 is allowed to see any of User1 presence information. ○ UE2 and UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has an active publication with all commonly supported Presence elements, some not allowed to be seen by anonymous users. ○ User2 has no active subscriptions. ○ User3 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User3 subscribes to Presence Information from User1 2. User2 subscribes as Anonymous to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE3 displays all presence information from User1. 2. UE2 displays Presence information of User1 according to the rules for anonymous subscriptions set by User1.

5.1.1.14 Presence-1.1-int-0122 Default Policy

Test Case Id	Presence-1.1-int-0122
Test Object	UEs with Presence Source, Presence Watcher and XDMC functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a User is able to define policies so that defined presence information can be sent to unspecified Users (not known in the Presence Rules document), acting as Watchers.</p> <p><u>TEST CASE GOAL:</u> Verify that a User, acting as a Presentity can define the contents an unspecified User, acting, as a Watcher will see.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information about default policy authorizations for presence information belonging to User1. User3 is allowed to see any of User1 presence information. ○ UE2 and UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ The default policy contains less information than all commonly supported Presence elements. ○ User1 has an active publication with all commonly supported Presence elements. ○ User2 has no active subscriptions. ○ User3 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User3 subscribes to presence information from User1. 2. User2 subscribes to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE3 displays Presence information of User1. 2. UE2 displays Presence information of User1 according to the default policy set by User1.

5.1.1.15 Presence-1.1-int-0123 Authorization management for groups

Test Case Id	Presence-1.1-int-0123
Test Object	UEs with Presence Source, Presence Watcher and XDMC functionality, Presence Server, and Shared XDMS.
Test Case Description	<p>Verify that a Presence Server can handle the Presence Rules document for groups of watchers stored in the Shared XDMS.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a XDMC, can modify his</p>

	permissions for groups of watcher stored in the Shared XDMS, and the PS handles these permissions properly.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1, User2 credentials) ○ Presence Server ○ Presence XDMS ○ Shared XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that Group A is authorized to see any of the presence information belonging to User1. ○ User2 is a member of Group A as defined by User1 ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User1 has the ability of changing the content of its Presence Authorization Rules Document. ○ User2 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported mandatory Presence elements. 2. User2 subscribe to presence information from User1. 3. User1 updates the Authorization Rules Document in PS XDMS to block Group A to see his presence. 4. User1 modifies the active presence information that has been published.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 display the presence information from User1. 4. UE2 does not display any presence information from User1.

5.1.1.16 Presence-1.1-int-0124 Combining permissions on an ongoing subscription

Test Case Id	Presence-1.1-int-0124
Test Object	UEs with Presence Source, Presence Watcher and XDMS functionality, Presence Server, and Shared XDMS.
Test Case Description	<p>Verify that a Presence Server can handle changes for the Presence Rules document for Watchers (individual Watchers or groups) stored in the Shared XDMS.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a XDMS, can modify his permissions for individual watchers and/or groups of watcher stored in the Shared XDMS, and the PS handles these permissions properly.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS ○ Shared XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 and User3, as members of a group stored in the Shared XDMS, are authorized to see any of the presence information belonging to User1. ○ UE2 and UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User1 has the ability of changing the content of its Presence Authorization Rules Document. ○ User2 has no active subscriptions. ○ User3 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 5. User1 publishes presence information for all commonly supported Presence elements. 6. User2 and User3 subscribe to presence information from User1. 7. User1 updates the Authorization Rules Document in PS XDMS to block User2 (as a regular watcher) to see his presence. 8. User1 modifies the active presence information that has been published. 9. User1 updates the Authorization Rules Document in PS XDMS to remove the block defined in step 3. 10. User1 modifies the active presence information that has been

	published. 11. User2 subscribes again to presence information from User1.
Pass-Criteria	3. UE2 and UE3 display the presence information from User1. 4. UE2 displays a blocking notification message, indicating that the User2 is not authorized to see User1's presence information 5. UE3 displays the updated presence information from User1. 6. UE3 display the updated presence information from User1 7. UE2 displays the updated presence information from User1.

5.1.1.17 Presence-1.1-int-0130 Publication of presence information, Watcher is blocked

Test Case Id	Presence-1.1-int-0130
Test Object	UE with presence source and presence Watcher functionality and XDMS, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that User1 successfully publishes presence information. User2 will not be able to Subscribe to the presence information when blocked by User1.</p> <p><u>TEST CASE GOAL:</u> Verify that when one user publishes presence information, another user, which is blocked, is not allowed to subscribe for presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is BLOCKED to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information ○ User1 and User2 will have a set of commonly supported Presence elements ○ User1 has no active publications. ○ User2 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 Publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information from User1. 3. User1 updates the Authorization Rules Document in PS XDMS to allow User2 to see his presence. 4. User2 subscribes to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays that User1 is not available. 4. UE2 displays the presence information published by User1

5.1.1.18 Presence-1.1-int-0140 Publication of presence information, Watcher is politely blocked

Test Case Id	Presence-1.1-int-0140
Test Object	UEs with presence source and presence Watcher functionality, Presence Server, Presence XDMS.
Test Case Description	User2 will be able to Subscribe and receive notifications, but presence information will not be revealed, since the user is politely blocked. <u>TEST CASE GOAL:</u> Verify that when one user publishes presence information, another user, acting as Watcher, will be able to subscribe and receive notification, but presence information will not be revealed, since the user is politely blocked.
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is POLITE BLOCKED to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the presence information of User1 as unavailable and unwilling to communicate.

5.1.1.19 Presence-1.1-int-0141 Notification Filtering (Includes Optional Features)

Test Case Id	Presence-1.1-int-0141
Test Object	UEs with Presence Source and Presence Watcher functionality with event notification filtering capabilities, Presence Server with event notification filtering capabilities.
Test Case Description	<p>Verify that a UE, acting as a Watcher, can subscribe using notification filtering and that the Presence Server applies filtering and notifies the correct information.</p> <p>TEST CASE GOAL: Verify that User2, acting as a Watcher, only gets the information that has been asked for.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information and setting a notification filter. ○ User1 and User2 have a set of commonly supported Presence elements. ○ User1 has an active publication with all commonly supported Presence elements. ○ User2 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User2 subscribes to presence information of User1 using a presence information filter for a subset of the commonly supported Presence elements.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE2 displays the resulting presence information according to the applied filter requested by User2.

5.1.1.20 Presence-1.1-int-0142 Event Notification Filtering (Includes Optional Features)

Test Case Id	Presence-1.1-int-0142
Test Object	UEs with Presence Source and Presence Watcher functionality with event notification filtering capabilities, Presence Server with event notification filtering capabilities.
Test Case Description	<p>Verify that a UE, acting as a Watcher, can subscribe using event notification filtering and that the Presence Server applies filtering and notifies the correct information whenever the filter is triggered.</p> <p>TEST CASE GOAL: Verify that User2, acting as a Watcher, only gets the information that has been asked for whenever the filter is triggered.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information and setting a notification filter. ○ User1 and User2 have a set of commonly supported Presence elements. ○ User1 has no active publication. ○ User2 has an active subscription to User 1 with a defined presence information filter and a defined triggering filter.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes some presence information that does not activate the triggering filter defined by User2. 2. User1 publishes some presence information that activates the triggering filter defined by User2.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE2 does not display any presence information from User1. 2. UE2 displays the resulting presence information according to the applied presence information filter requested by User2.

5.1.1.21 Presence-1.1-int-0143 Notification of Only Authorized Presence Information is sent to Watchers (Includes Optional Features)

Test Case Id	Presence-1.1-int-0143
Test Object	UE with Presence Source and UE with Presence Watcher functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a watcher is notified of a specified subset of the presence information of a presentity, if the watcher falls into a group that the presentity decides to reveal a subset of his/her presence information to.</p> <p><u>TEST CASE GOAL:</u> Verify that UE2 and UE3 display the information User1 authorizes them to see. Verify that UE2 displays only the information UE1 authorizes for the group of which User2 is a member.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2, and User3 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see all of the presence information belonging to User1, and User2 is authorized to see a subset of the presence information belonging to User1. ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1. ○ In the Presence XDMS, the Presence Authorization Rules document contains information that a group (e.g. Group1) that User2 belongs to is authorized to see a subset of the presence information belonging to User1. ○ UE2 and UE3 are capable of displaying a set of commonly supported elements. NOTE: The presence information for User2 is a subset of the presence information for User3. ○ UE2 and UE3 are capable of displaying the same presence information. ○ User1, User2, and User3 have a set of commonly supported Presence elements. ○ User1 has no active publications ○ User2 and User3 have no active subscriptions for User1
Test Procedure	<ol style="list-style-type: none"> 1. User2 and User3 subscribe to presence information from User1. 2. User1 publishes its presence information for all commonly supported of its Presence elements.

Pass-Criteria	2a. UE3 displays all of the commonly supported presence information published by User1 2b. UE2 displays a subset of the commonly supported presence information published by User1 that User2 is authorized to see
----------------------	---

5.1.1.22 Presence-1.1-int-0144 Reactive Authorization for a specific group (Includes Optional Features)

Test Case Id	PS-1.1-int-0144
Test Object	UE with Presence Source and UE with Presence Watcher functionality, Presence Server.
Test Case Description	<p>Verify that a presentity can authorize a group of watchers to subscribe to his/her presence information when the request from that watcher arrives (Reactive Authorization).</p> <p><u>TEST CASE GOAL:</u> Verify that UE1(the Presentity) can Reactively Authorize a group of watchers of which UE2 is a member..</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presences XDMS, the Presence Authorization Rules document contains information that a group (e.g. Group1) that User2 belongs to is authorized to see any the presence information belonging to User1. ○ UE1 is capable of watcher information subscriptions. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has no active publications ○ User2 has no active subscriptions for User1
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User2 subscribes to presence information for User1 3. UE1 displays and User1 accepts a request to allow Group1 to get presence information
Pass-Criteria	<ol style="list-style-type: none"> 3. UE2 displays the presence information published by User1

5.1.2 Error Flow

5.1.2.1 Presence-1.1-int-0145 Publication of Presence information not supported by watcher

Test Case Id	Presence-1.1-int-0145
Test Object	UE with Presence Source and UE with Presence Watcher functionality, Presence Server.
Test Case Description	<p>Verify that presence information with non-OMA elements published by an UE will be handled correctly by another UE, which subscribes for that information.</p> <p><u>TEST CASE GOAL:</u> Verify that when UE1 publishes presence information with non-OMA elements, UE2, as Watcher, will interpret correctly the OMA presence information elements and discard the non-OMA ones.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE2 capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. User1 also supports pdf extensions not defined in OMA. ○ User1 has no active publications ○ User2 has no active subscriptions for User1
Test Procedure	<ol style="list-style-type: none"> 1. User2 subscribes to presence information from User1. 2. User1 publishes presence information for all his supported Presence elements.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE2 displays the commonly supported presence elements.

5.2 Resource List Test Cases

5.2.1 Normal Flow

5.2.1.1 Presence-1.1-int-0150 Subscription to a resource list

Test Case Id	Presence-1.1-int-0150
Test Object	UEs with Presence Source, Presence Watcher, Presence Server, RLS and RLS XDMS.
Test Case Description	<p>Verify that an RLS can handle subscriptions to resource lists and distribute notifications including presence information to the Watcher.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a Watcher, is able to subscribe to a resource list with URI's to Presence Sources and be able to display presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Resource List Server ○ Presence XDMS ○ RLS XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ A resource list is stored in RLS XDMS with URI's to User1 and User2. ○ UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active publications. ○ User3 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 and User 2 publish presence information for all commonly supported Presence elements. 2. User3 subscribes to its resource list stored in RLS XDMS
Pass-Criteria	<ol style="list-style-type: none"> 2. UE3 displays the presence information from User1 and User2.

5.2.1.2 Presence-1.1-int-0151 Adding a Presentity to an ongoing list subscription

Test Case Id	Presence-1.1-int-0151
Test Object	UEs with Presence Source, Presence Watcher functionality and XDMC, Presence Server, RLS and RLS XDMS.
Test Case Description	<p>Verify that an RLS can handle subscriptions to resource lists and distribute notifications including presence information to the Watcher.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a Watcher, is able to display presence information for a specific user, which is added to a resource list the Watcher owns and has an active subscription on.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Resource List Server ○ Presence XDMS ○ RLS XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ UE3 capable of displaying presence information. ○ User3 has the ability of adding user references to its Resource List stored in RLS XDMS. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active publications. ○ User3 has an active subscription to a list, which contains a reference to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. A reference to User2 is added to the resource list of which User3 has an active subscription. 3. User2 publishes presence information for all commonly supported Presence elements.
Pass-Criteria	<ol style="list-style-type: none"> 1. UE3 displays the presence information published by User1. 3. UE3 displays the presence information published by User2.

5.2.1.3 Presence-1.1-int-0152 Subscription to shared lists

Test Case Id	Presence-1.1-int-0152
Test Object	UEs with Presence Source, Presence Watcher, Presence Server, Resource List Server, RLS XDMS and Shared XDMS.
Test Case Description	<p>Verify that an RLS can handle subscriptions to resource lists pointing to groups in the Shared XDMS, and distribute notifications including presence information to the Watcher.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a Watcher, is able to subscribe to resource lists pointing to groups in the Shared XDMS and be able to display presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Resource List Server ○ Presence XDMS ○ RLS XDMS ○ Shared XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ A resource list with a pointer to a group in the Shared XDMS is stored in RLS XDMS. This group in the Shared XDMS contains URI's to User1 and User2. ○ UE3 capable of displaying presence information. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active publications. ○ User3 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 and User2 publish presence information for all commonly supported Presence elements. 2. User3 subscribes to its resource list stored in RLS XDMS
Pass-Criteria	<ol style="list-style-type: none"> 2. UE3 displays the presence information from User1 and User2.

5.2.1.4 Presence-1.1-int-0153 Adding a Presentity to an ongoing shared list subscription

Test Case Id	Presence-1.1-int-0153
Test Object	UEs with Presence Source, Presence Watcher functionality and XDMS, Presence Server, RLS, RLS XDMS and Shared XDMS.
Test Case Description	<p>Verify that an RLS can handle subscriptions to resource lists pointing to groups in the Shared XDMS and distribute notifications including presence information to the Watcher.</p> <p><u>TEST CASE GOAL:</u> Verify that a UE, acting as a Watcher, is able to display presence information for a specific user, which is added to a shared group pointed by a resource list the Watcher owns and has an active subscription on.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Resource List Server ○ Presence XDMS ○ RLS XDMS ○ Shared XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ UE3 capable of displaying presence information. ○ User3 has the ability of adding user references to its URI List stored in the Shared XDMS and referenced by its Resource List stored in RLS XDMS. ○ User1, User2 and User3 will have a set of commonly supported Presence elements. ○ User1 has no active publications. ○ User2 has no active publications. ○ User3 has an active subscription to a list, which contains a reference to User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. Users3 adds User2 to the URI list of which User3 has an active subscription. 3. User2 publishes presence information for all commonly supported Presence elements.

Pass-Criteria	<ol style="list-style-type: none"> 1. UE3 displays the presence information published by User1. 3. UE3 displays the presence information published by User2.
---------------	--

5.2.1.5 Presence-1.1-int-0160 RLS Event notification filtering (Includes Optional Features)

Test Case Id	Presence-1.1-int-O-0160
Test Object	UEs with Presence Source and Presence Watcher functionality with event notification filtering capabilities, Presence Server with event notification filtering capabilities, RLS with event notification filtering capabilities.
Test Case Description	<p>Verify that a UE, acting as a Watcher, can subscribe using event notification filtering to a resource list and that the Resource List Server applies filtering and notifies the correct information whenever the filter is triggered.</p> <p>TEST CASE GOAL: Verify that an RLS can handle event and content subscriptions to resource lists and distribute notifications to the Watcher including appropriate presence information whenever the filter is triggered.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 3 UEs (with User1, User2 and User3 credentials) ○ Presence Server ○ Presence XDMS ○ RLS ○ RLS XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User3 is authorized to see any of the presence information belonging to User1 and User2. ○ UE3 is capable of displaying presence information and setting a notification filter. ○ User1, User2 and User3 have a set of commonly supported Presence elements. ○ User1 and User2 have no active publications. ○ User3 has no active subscriptions to User1 and User2. ○ A resource list for User3 is stored in RLS XDMS with URI's to User1 and User2.
Test Procedure	<ol style="list-style-type: none"> 1. User3 subscribes to its resource list with a defined presence information filter and a defined triggering filter. 2. User1 publishes some presence information that does not activate the triggering filter defined by User3. 3. User2 publishes some presence information that activates the

	triggering filter defined by User3.
Pass-Criteria	<ol style="list-style-type: none">1. UE3 does not display any presence information from User1 and User2.2. UE3 does not display any presence information from User1 and User2.3. UE3 displays the resulting presence information for User2 according to the applied presence information filter requested by User3.

5.2.2 Error Flow

Not Available

5.3 Watcher Information Subscriber Test Cases

5.3.1 Normal Flow

5.3.1.1 Presence-1.1-int-0200 Subscribe to Watcher Information

Test Case Id	Presence-1.1-int-0200
Test Object	UEs with Presence Source, Presence Watcher and Watcher Information Subscriber functionality, Presence Server
Test Case Description	Verify that a UE successfully Subscribes to Watcher Information TEST CASE GOAL: User1 subscribes to watcher information and will be notified when User2 subscribes to User1's presence information
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE1 is capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has an active publication with all commonly supported Presence elements. ○ User2 has no active subscriptions for User1.
Test Procedure	<ol style="list-style-type: none"> 1. User1 subscribes to Watcher Information Subscriber. 2. User2 subscribes to presence information from User1.
Pass-Criteria	<ol style="list-style-type: none"> 2. UE1 displays that User2 has subscribed to User1's presence information. UE2 displays presence information published by User1.

5.3.1.2 Presence-1.1-int-0210 Publication of presence information, Watcher is pending

Test Case Id	Presence-1.1-int-0210
Test Object	UEs with Presence Source, Presence Watcher and Watcher Information Subscriber functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that UE successfully publishes and receives presence information after reactive authorization.</p> <p><u>TEST CASE GOAL:</u> Verify that User1 successfully can subscribe for watcher information and is notified when User2 subscribes for User1's presence information. User1 then updates his Authorization Rules Document to allow User2 to see his presence. User2 will display User1's presence information.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS ○ XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2's policy is CONFIRM (reactive authorization) to see any of the presence information belonging to User1. ○ UE2 is capable of displaying presence information. ○ User1 and User2 will have a set of commonly supported Presence elements. ○ User1 has the ability of changing the content of its Presence Authorization Rules Document. ○ User1 has no active publications. ○ User2 has no active subscriptions.
Test Procedure	<ol style="list-style-type: none"> 1. User1 publishes presence information for all commonly supported Presence elements. 2. User1 subscribes to "Watcher Information Subscriber". 3. User2 subscribes to presence information from User1. 4. User1 updates the Authorization Rules Document in PS XDMS to allow User2 to see his presence.
Pass-Criteria	<ol style="list-style-type: none"> 3. UE1 displays that User2 has requested to see his presence information. 4. UE2 displays the presence information from User1.

5.3.1.3 Presence-1.1-int-0211 Notification to Watcher Information Subscriber when a subscription to His/Her Presence Information Expires (Includes Optional features)

Test Case Id	Presence-1.1-int-0211
Test Object	UEs with Presence Source, Watcher and Watcher Information Subscriber functionality, Presence Server, Presence XDMS.
Test Case Description	<p>Verify that a Watcher Information Subscriber can receive notifications whenever one of the subscription to his/her Presence Information expires.</p> <p><u>TEST CASE GOAL:</u> Verity that User1 (Watcher Information Subscriber) can get notifications whenever User2's subscription to User1's Presence Information expires.</p>
Specification Reference	Refer to Appendix A
SCR Reference	Refer to Appendix A
Tool	N/A
Test Code	N/A
Preconditions	<ul style="list-style-type: none"> • Equipment: <ul style="list-style-type: none"> ○ 2 UEs (with User1 and User2 credentials) ○ Presence Server ○ Presence XDMS • Prerequisite for this test: <ul style="list-style-type: none"> ○ In the Presence XDMS, the Presence Authorization Rules document contains information that User2 is authorized to see any of the presence information belonging to User1. ○ UE1 is capable of watcher information subscriptions. ○ UE1 is capable of displaying Watcher Information. ○ UE2 is capable of displaying presence information. ○ User1 and User2 have a set of commonly supported mandatory Presence elements. ○ User1 has an active publication with all commonly supported Presence elements. ○ User2 has no active subscriptions for User1.
Test Procedure	<ol style="list-style-type: none"> 3. User1 subscribes to Watcher Information. 4. User2 subscribes to presence information from User1. 5. User2's subscription expires (e.g. removing battery).
Pass-Criteria	<ol style="list-style-type: none"> 2. UE1 displays that User2 has subscribed to User1's presence information. UE2 displays the presence information published by User1. 3. User1 displays that User2 subscription to User1 presence information has expired.

5.3.2 Error Flow

Not available.

Appendix A. Change History (Informative)

A.1 Approved Version History

Reference	Date	Description
n/a	n/a	No prior version –or- No previous version within OMA

A.2 Draft/Candidate Version 1.1 History

Document Identifier	Date	Sections	Description
Draft Version OMA-ETS-Presence_SIMPLE_INT-V1_1	04 Dec 2007	all	Initial version based on OMA-ETS-Presence_SIMPLE_INT-V1_0-20060606-C, References to PRS 1.0 specifications changed to PRS 1.1 and template updated
Candidate Version OMA-ETS-Presence_SIMPLE_INT-V1_1	31 Mar 2008	n/a	TP approval document reference: OMA-TP-2008-0108

Appendix B. SCR and Specification References

Test Case Number in ETS	SCR-reference	Spec (AD,CP,UP)-reference
Presence-1.0-int-0100 Presence-1.0-int-0101 Presence-1.0-int-0102 Presence-1.0-int-0104 Presence-1.0-int-0105 Presence-1.0-int-0107 Presence-1.0-int-0120 Presence-1.0-int-0122 Presence-1.0-int-0130 Presence-1.0-int-0145	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.7
Presence-1.0-int-0106	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-004:O SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 5.1.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.1.2 5.4.2 5.7
Presence-1.0-int-0108	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-PS-S-021:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-005:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.2.4 5.4 5.4.1 5.4.2 5.4.2.2 5.7

	SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	
Presence-1.0-int-109	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O Presence_SIMPLE-XDMC-C-001:M Presence_SIMPLE-XDMC-C-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.6 5.7
Presence-1.0-int-0110 Presence-1.0-int-111	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.6 5.7
Presence-1.0-int-121 Presence-1.0-int-122 Presence-1.0-int-124	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.6 5.7

	Presence_SIMPLE-XDMC-C-001:M Presence_SIMPLE-XDMC-C-002:O	
Presence-1.0-int-0123	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:M SIMPLE-PS-S-013:O SIMPLE-PS-S-014:M SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.1.2 5.4.2 5.7
Presence-1.0-int-140	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-PS-S-022:M SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.7
Presence-1.0-int-141 Presence-1.0-int-142	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2

PS-1.0-int-0143	[OMATSPRES] SIMPLE-PS-S-012:M SIMPLE-PS-S-013:O SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M Presence_SIMPLE-PresenceXDMS-S-001:M IMPL-RC-C-003:M SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M Presence_SIMPLE-XDMC-C-001:M	[OMATSPRES]: 5.1.1 5.2.1 5.3 5.4.1 5.4.2 5.6 10.1
PS-1.0-int-0144	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-PS-S-019:M SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O SIMPLE-WIS-C-001:O SIMPLE-WIS-C-002:O SIMPLE-WIS-C-003:O SIMPLE-WIS-C-004:O SIMPLE-WIS-C-005:O	[OMATSPRES]: 5.4.3.2 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.7
Presence-1.0-int-150	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-003:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O SIMPLE-RLS-S-001:M SIMPLE-RLS-S-002:M Presence_SIMPLE-RLSXDMS-S-001:M Presence_SIMPLE-RLSXDMS-S-002:O SIMPLE-RLS-C-001:M SIMPLE-RLS-C-002:M	[OMATSPRES]: 5.1.1 5.2.1 10.1 10.4 5.4 5.4.1 5.4.2 5.5 5.7 5.8

Presence-1.0-int-151 Presence-1.0-int-152 Presence-1.0-int-153	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-003:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O SIMPLE-RLS-S-001:M SIMPLE-RLS-S-002:M Presence_SIMPLE-RLSXDMS-S-001:M Presence_SIMPLE-RLSXDMS-S-002:O Presence_SIMPLE-XDMC-C-001:M Presence_SIMPLE-XDMC-C-002:O SIMPLE-RLS-C-001:M SIMPLE-RLS-C-002:M	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.5 5.6 5.7 5.8
Presence-1.0-int-160	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O SIMPLE-RLS-S-001:M SIMPLE-RLS-S-002:M Presence_SIMPLE-RLSXDMS-S-001:M Presence_SIMPLE-RLSXDMS-S-002:O SIMPLE-RLS-C-001:M SIMPLE-RLS-C-002:M	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.4 5.4.1 5.4.2 5.6 5.7 5.8
Presence-1.0-int-200 Presence-1.0-int-211	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.3 5.4 5.4.1 5.4.2

	SIMPLE-PS-S-012:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O SIMPLE-WIS-C-001:O SIMPLE-WIS-C-002:O SIMPLE-WIS-C-003:O SIMPLE-WIS-C-004:O SIMPLE-WIS-C-005:O	5.4.4 5.7
Presence-1.0-int-210	SIMPLE-SRC-C-001:M SIMPLE-SRC-C-002:M SIMPLE-SRC-C-003:M SIMPLE-SRC-C-010:M SIMPLE-SRC-C-011:O SIMPLE-PS-S-001:M SIMPLE-PS-S-002:M SIMPLE-PS-S-003:M SIMPLE-PS-S-004:M SIMPLE-PS-S-012:O SIMPLE-PS-S-013:O SIMPLE-WATCH-C-001:M SIMPLE-WATCH-C-002:M SIMPLE-WATCH-C-004:M SIMPLE-WATCH-C-007:O SIMPLE-WATCH-C-008:O Presence_SIMPLE-PresenceXDMS-S-001:M Presence_SIMPLE-PresenceXDMS-S-002:O SIMPLE-WIS-C-005:O Presence_SIMPLE-XDMC-C-001:M Presence_SIMPLE-XDMC-C-002:O	[OMATSPRES]: 5.1.1 10.1 10.4 5.2.1 5.3 5.4 5.4.1 5.4.2 5.6 5.7