ETSI TS 124 141 V7.2.0 (2006-09)

Technical Specification

Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Presence service using the IP Multimedia (IM) Core Network (CN) subsystem; Stage 3 (3GPP TS 24.141 version 7.2.0 Release 7)

Reference

RTS/TSGC-0124141v720

Keywords

GSM, UMTS

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under http://webapp.etsi.org/key/queryform.asp .

Contents

Foreword

This Technical Specification has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document provides the protocol details for the presence service within the IP Multimedia (IM) Core Network (CN) subsystem based on the Session Initiation Protocol (SIP) and SIP Events as defined in 3GPP TS 24.229 [9].

Where possible the present document specifies the requirements for this protocol by reference to specifications produced by the IETF within the scope of SIP and SIP Events, either directly, or as modified by 3GPP TS 24.229 [9].

Requirements for manipulation of presence data are defined by use of a protocol at the Ut reference point based on XML Configuration Access Protocol (XCAP) (draft-ietf-simple-xcap [33]).

The present document is applicable to Application Servers (ASs) and User Equipment (UE) providing presence functionality.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document in the same Release as the present document.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 22.141: "Presence Service; Stage 1".

[3] 3GPP TS 23.002: "Network architecture".

[4] 3GPP TS 23.141: "Presence service; Architecture and functional description; Stage 2".

[5] 3GPP TS 23.218: "IP Multimedia (IM) session handling; IM call model; Stage 2".

[6] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[7] 3GPP TS 24.109: "Bootstrapping interface (Ub) and Network application function interface (Ua); Protocol details".

[8] 3GPP TS 24.228 Release 5: "Signalling flows for the IP multimedia call control based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".

[9] 3GPP TS 24.229: "Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3".

[10] 3GPP TS 29.228: "IP Multimedia (IM) Subsystem Cx and Dx Interfaces; Signalling flows and message contents".

[11] 3GPP TS 33.222: "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[12] IETF RFC 2664 (1999): "FYI on Questions and Answers - Answers to Commonly asked New Internet User Questions".

[13] IETF RFC 2246 (1999): "The TLS Protocol Version 1.0".

[14] IETF RFC 2387 (August 1998): "The MIME Multipart/Related Content-type".

[15] IETF RFC 2616 (June 1999): "Hypertext Transfer Protocol -- HTTP/1.1".

[15A] IETF RFC 2617 (June 1999): " HTTP Authentication: Basic and Digest Access Authentication".

[16] IETF RFC 2778 (2000): "A Model for Presence and Instant Messaging".

[17] IETF RFC 3261 (June 2002): "SIP: Session Initiation Protocol".

[18] IETF RFC 3263 (June 2002): "Session Initiation Protocol (SIP): Locating SIP Servers".

[19] IETF RFC 3265 (March 2002): "Session Initiation Protocol (SIP)-Specific Event Notification".

[20] IETF RFC 3310 (2002): "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)".

[21] IETF RFC 3863 (August 2004): "Presence Information Data Format (PIDF)".

[22] draft-ietf-simple-event-list-07 (December 2004): "A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[23] IETF RFC 3903 (October 2004): " Session Initiation Protocol (SIP) for Event State Publication".

[24] draft-ietf-simple-partial-notify-06 (October 2005): " Session Initiation Protocol (SIP) extension for Partial Notification of Presence Information".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[25] draft-ietf-simple-prescaps-ext-05 (October 2005): "User Agent Capability Extension to Presence Information Data Format (PIDF)".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[26] draft-ietf-simple-rpid-09 (September 2005): "RPID: Rich Presence Extensions to the Presence Information Data Format (PIDF)".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[27] IETF RFC 3856 (August 2004): "A Presence Event Package for the Session Initiation Protocol (SIP)".

[28] IETF RFC 3857 (August 2004): "A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)".

[29] IETF RFC 3858 (August 2004): "An Extensible Markup Language (XML) Based Format for Watcher Information".

[30] draft-ietf-simple-filter-format-05 (March 2005): "An Extensible Markup Language (XML) Based Format for Event Notification Filtering".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[31] draft-ietf-simple-event-filter-funct-05 (March 2005): "Functional Description of Event Notification Filtering".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[32] draft-ietf-simple-cipid-06 (July 2005): "CIPID: Contact Information in Presence Information Data Format".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[33] draft-ietf-simple-xcap-08 (October 2005): "The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[34] draft-ietf-simple-xcap-pidf-manipulation-usage-02 (October 2004): "An Extensible Markup Language (XML) Configuration Access Protocol (XCAP) Usage for Manipulating Presence Document Contents".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[35] draft-ietf-simple-presence-rules-04 (October 2005): "Presence Authorization Rules".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[36] draft-ietf-simple-xcap-list-usage-05 (February 2005): "An Extensible Markup Language (XML) Format for Representing Resource Lists".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[37] draft-ietf-geopriv-pidf-lo-03 (September 2004): "A Presence-based GEOPRIV Location Object Format".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[38] draft-ietf-simple-partial-pidf-format-05 (October 2005): "Presence Information Data Format (PIDF) Extension for Partial Presence".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[39] draft-ietf-simple-xcap- diff-02 (October 2005): "An Extensible Markup Language (XML) Document Format for Indicating Changes in XML Configuration Access Protocol (XCAP) Resources".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[40] draft-ietf-sip-content-indirect-mech-05 (October 2004): "A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[41] draft-ietf-simple-common-policy-caps-00 (July 2005): "An Extensible Markup Language (XML) Representation for Expressing Policy Capabilities".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[42] draft-ietf-simple-pres-policy-caps-00 (July 2005): "An Extensible Markup Language (XML) Representation for Expressing Presence Policy Capabilities".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[43] draft-ietf-sipping-config-framework-07 (July 2005): "A Framework for Session Initiation Protocol User Agent Profile Delivery".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[44] draft-ietf-simple-presence-data-model-07 (January 2006): "A Data Model for Presence".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

[45] draft-ietf-simple-partial-publish-03 (July 2005): "Publication of Partial Presence Information".

Editor's note: The above document cannot be formally referenced until it is published as an RFC.

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TR 21.905 [1] and the following apply:

subscription authorization policy: a policy that determines which watchers are allowed to subscribe to diffa presentity's presence information
The subscription authorization policy also determines to which presentity's presence information the watcher has access.

For the purposes of the present document, the following terms and definitions given in 3GPP TS 23.141 [4] apply:

Presence list server
Presence Network Agent (PNA)
Presence Server (PS)
Presence User Agent (PUA)

For the purposes of the present document, the following terms and definitions from RFC 2778 [16] apply:

Presence tuple
Presentity

For the purposes of the present document, the following terms and definitions from RFC 3903 [23] apply:

Event Publication Agent (EPA)
Event State Compositor (ESC)

For the purposes of the present document, the following terms and definitions from draft-ietf-simple-xcap [33] apply:

XCAP client
XCAP server

For the purposes of the present document, the following terms and definitions from draft-ietf-simple-event-list [22] apply:

Resource List Server (RLS)

For the purposes of the present document, the following terms and definitions given in RFC 1594 [12].

Fully-Qualified Domain Name (FQDN)

For the purposes of the present document, the following terms and definitions given in RFC 3261 [17] apply:

Final response
Header
Header field
Method
Request
Response
(SIP) transaction
Status-code (see RFC 3261 [17], subclause 7.2)
Tag (see RFC 3261 [17], subclause 19.3)

For the purposes of the present document, the following terms and definitions given in 3GPP TS 23.002 [3], subclauses 4.1.1.1 and 4a.7 apply:

Call Session Control Function (CSCF)
Home Subscriber Server (HSS)

For the purposes of the present document, the following terms and definitions given in 3GPP TS 23.218 [5], subclause 3.1 apply:

Filter criteria
Initial filter criteria
Subsequent request

For the purposes of the present document, the following terms and definitions given in 3GPP TS 23.228 [6], subclauses 4.3.3.1 and 4.6 apply:

Interrogating-CSCF (I-CSCF)
Proxy-CSCF (P-CSCF)
Serving-CSCF (S-CSCF)

For the purposes of the present document, the following terms and definitions given in 3GPP TR 21.905 [1] apply:

User Equipment (UE)

For the purposes of the present document, the following terms and definitions from 3GPP TS 33.222 [11] apply:

Authentication Proxy

3.2 Abbreviations

For the purposes of the present document, the following abbreviations apply:

AS	Application Server
AUID	Application Unique ID
CN	Core Network
CPIM	Common Profile for Instant Messaging
CSCF	Call Session Control Function
EPA	Event Publication Agent
ESC	Event State Compositor
HSS	Home Subscriber Server
HTTP	HyperText Transfer Protocol
I-CSCF	Interrogating - CSCF
IM	IP Multimedia
IOI	Inter Operator Identifier
IP	Internet Protocol
MIME	Multipurpose Internet Mail Extensions
P-CSCF	Proxy - CSCF
PIDF	Presence Information Data Format
PNA	Presence Network Agent
PS	Presence Server
PSI	Public Service Identity
PUA	Presence User Agent
RLMI	Resource List Meta-Information
RLS	Resource List Server
RPID	Rich Presence Information Data
S-CSCF	Serving - CSCF
SIP	Session Initiation Protocol
TLS	Transport Layer Security
UE	User Equipment
URI	Universal Resource Identifier
XCAP	XML Configuration Access Protocol
XML	Extensible Markup Language

4 Presence service overview

The presence service provides the ability for the home network to manage presence information of a user's device, service or service media even whilst roaming. A user's presence information may be obtained through input from the user, information supplied by network entities or information supplied by elements external to the home network. Consumers of presence information, watchers, may be internal or external to the home network. The architecture for the 3GPP presence service is specified in 3GPP TS 23.141 [4].

SIP and XCAP provide means to manipulate the presence status of a user. For details on the differences between those means refer to RFC 3903 [23] and draft-ietf-simple-xcap-pidf-manipulation-usage [34]. For details on the relationship of XCAP server to other roles see subclause 6.2.2.

5 SIP related procedures

5.1 Introduction

5.2 Functional entities

5.2.1 User Equipment (UE)

A UE shall implement the role of a PUA (see subclause 5.3.1), a watcher (see subclause 5.3.2) or both.

5.2.2 Application Server (AS)

An AS may implement one or more of the roles of a PUA (see subclause 5.3.1), watcher (see subclause 5.3.2), PS (see subclause 5.3.3), RLS (see subclause 5.3.4), or PNA (see subclause 5.3.5).

For this version of the present document, the interface between the PNA and the PS is not defined.

5.3 Roles

5.3.1 Presence User Agent (PUA)

5.3.1.1 General

A PUA is an entity that provides presence information to a PS.

In addition to the procedures specified in subclause 5.3.1, the PUA shall support the procedures specified in 3GPP TS 24.229 [9] appropriate to the functional entity in which the PUA is implemented.

5.3.1.2 Publication of presence information

When the PUA intends to publish its own view of the presentity's presence information, it shall generate a PUBLISH request by acting as an Event Publication Agent (EPA) in accordance with RFC 3903 [23].

Editor's note: The details of generating presence document are for further study. This material was originally contained within draft-ietf-simple-presence-data-model-07, which was referenced by this specification, but was removed from a later version of the draft. Current work is contained in draft-rosenberg-simple-presence-processing-model.

The PUA shall implement the "application/pidf+xml" content type as described in RFC 3863 [21] ,the Presence Information Data Format (PIDF) extensions defined in draft-ietf-simple-rpid [26].

The PUA may implement the PIDF extensions defined in draft-ietf-simple-cipid [32].

The PUA may implement location information according to the format defined in draft-ietf-geopriv-pidf-lo [37].

NOTE 1: The categorization of presence attributes to generic information attributes and communication address specific attributes is done using the <person> and <tuple> elements as defined in draft-ietf-simple - presence-data-model [44].

The PUA shall implement draft-ietf-simple-prescaps-ext [25] if it wants to make use of SIP user agent capabilities in the presence document. The extension may be used for describing the type of the service described by the presence tuple.

The PUA may implement draft-ietf-simple-partial-publish [45] if it wants to use the partial publication mechanism. The first partial PUBLISH request shall contain the full state. The PUA uses the "application/pidf-diff+xml" content type as described in draft-ietf-simple-partial-pidf-format [38] .

The PUA shall update the presence information, either 600 s before the publication expiration time if the publication period indicated from the PS in the response to the PUBLISH request was for greater than 1 200 s, or when half of the time has expired if the publication period was for 1 200 s or less, unless the PUA has determined that an update to the presence information is not required.

When the PUA intends to show different value of the same presence attribute to different watchers, the PUA shall publish a tuple or person element for every value it intends to show, all including a different value of the same presence attribute. The PUA shall label different information with different value of the <class> element in every published tuple or person element as defined in draft-ietf-simple-rpid [26]. The PUA shall also authorize different tuples to different watchers or watcher groups by manipulating the subscription authorization policy as defined in subclause 6.3.1.2.

If a local configuration information limiting the rate at which PUA is allowed to generate PUBLISH requests is available, then PUA shall take that information into account. Such local configuration information could be e.g. the shortest time period between consecutive PUBLISH requests.

5.3.1.3 Mapping of presence attributes

The eXtensible Markup Language (XML) Schema Definition of the "application/pidf+xml" format covers the definition of the 3GPP subscriber's presence attributes and the PUA shall perform the following mapping:

- the communication address (containing communication means, status and contact address) attribute and the priority attribute are represented by a <tuple> element including a basic <status> element and a <contact> elements containing a priority attribute as defined in RFC 3863 [21].

  The PUA represents subscriber specific information by including a <person> element defined in draft-ietf-simple-presence-data-model [44] . The person element may contain e.g. <activities> and <place-type> elements both defined in draft-ietf-simple-rpid [26]. Further PIDF extensions as defined in draft-ietf-simple-cipid [32] can also be used.

NOTE 1: draft-ietf-simple-presence-data-model [44] defines also a <device> element which can be used to present device specific information.

- the text attribute is represented by the <note> element as defined in RFC 3863 [21] for <tuple> elements and in draft-ietf-simple-presence-data-model [44] for <person> and <device> elements; and

- the location attribute is represented by the elements defined in draft-ietf-geopriv-pidf-lo [37] and the <place-type> element defined in draft-ietf-simple-rpid [26].

NOTE 2: Only information elements either relevant for the application or recommended by the presence –data model draft-ietf-simple-presence-data-model [44] are included in the PUBLISH request. Attributes not relevant or available (e.g. the text attribute or the location attribute) are omitted.

Additional extensions can be used to express application specific attributes, but their usage is outside the scope of this version of the specification.

5.3.1.4 Storing presence attributes by multipart/related or content indirection

The PUA shall implement the "multipart/related" content type as described in RFC 2387 [14] if it wants to aggregate other Multipurpose Internet Mail Extensions (MIME) objects with the "application/pidf+xml" content type.

When a presence attribute has a value of a MIME object, the PUA shall either:

a) publish the presence document and the MIME object utilizing the "multipart/related" content-type in the PUBLISH request; or

b) make use of content indirection.

When the PUA decides to use the content indirection mechanism for publishing an initial or modified value of a presence attribute the PUA shall follow the following procedure:

a) either store the MIME object behind an HTTP URI on the PS or ensure that the MIME object and a HTTP URL pointing to that MIME object already exists on the PS;

b) use the "multipart/related" content type as described in RFC 2387 [14] with the content indirection mechanism as specified in draft-ietf-sip-content-indirect-mech [40] for the publication of presence information format as follows:

  - set a CID URI referencing to other MIME multipart body. The other multipart body contains the content indirection information which is represented as the value of an XML element;

  - include the presence document of the format "application/pidf+xml" or "application/pidf-diff+xml" in the root of the body of the "multipart/related" content;

  - specify the part having information about the MIME object by using the "message/external-body" content type, defining the HTTP URI, versioning information and other information about the MIME object as described in draft-ietf-sip-content-indirect-mech [40].

NOTE 1: The versioning information is used for determining whether or not the MIME object indirectly referenced by a URI has changed or not;

When storing a MIME object on the PS the PUA shall:

a) construct as many HTTP URIs as the number of objects to be stored; and

b) formulate every HTTP URI according to a predefined directory structure.

NOTE 2: The PUA has the root directory for storing the MIME objects on the PS preconfigured.

NOTE 3: The PUA needs to store the MIME objects on the PS behind the HTTP URI(s) created previously using standard HTTP procedures as defined in RFC 2616 [15].

5.3.1.5 Subscription for the watcher information event template package

Upon activation of the presence service, the PUA application may subscribe for the watcher information state changes in accordance with RFC 3857 [28] and RFC 3858 [29].

The PUA application may include filters in the body of the SUBSCRIBE request in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

5.3.1.6 Subscription for notification of state changes in XML document

In order to get notifications of changes to XML documents manipulated via the Ut reference point the PUA may generate a SUBSCRIBE request in accordance with draft-ietf-simple-xcap-diff [39] and draft-ietf-sipping-config-framework [43].

5.3.2 Watcher

5.3.2.1 General

A watcher is an entity that subscribes to or requests presence information about a presentity from the PS.

In addition to the procedures specified in subclause 5.3.2, the watcher shall support the procedures specified in 3GPP TS 24.229 [9] appropriate to the functional entity in which the watcher is implemented.

5.3.2.2 Subscription for presence information state changes and notification acceptance

When the watcher intends to subscribe for presence information state changes of a presentity, it shall generate a SUBSCRIBE request in accordance with RFC 3265 [19] and RFC 3856 [27].

The watcher shall implement the "application/pidf+xml" content type as described in RFC 3863 [21] , the PIDF extensions defined in draft-ietf-simple-rpid [26].

The watcher may implement the PIDF extensions defined in draft-ietf-simple-cipid [32].

The watcher may implement location information according to the format defined in draft-ietf-geopriv-pidf-lo [37].

The watcher shall implement draft-ietf-simple-prescaps-ext [25] if it wants to make use of SIP user agent capabilities extensions included in the presence document. The extension may be used by the watcher for interpreting the type of the service described by the presence tuple.

The watcher may include filters in the body of the SUBSCRIBE request in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

The watcher may indicate its support for partial notification using the Accept header field in accordance with draft-ietf-simple-partial-notify [24].

The watcher shall interpret the received presence information according to draft-ietf-simple-presence-data-model [44] and the following:

a) a <person> element as defined in draft-ietf-simple-presence-data-model [44] means information about the presentity;

b) a tuple including a <relationship> element defined in draft-ietf-simple-rpid [26] means information about an alternate contact to the presentity;

c) a tuple contains communication means specific information. The communication means described by the tuple is deduced from the URI scheme of the contact address information present in the <contact> element as defined in RFC 3863 [21]. If the URI scheme of the contact address information provides ambiguous information about the communication means, the watcher shall further examine other elements of the tuple to decide the communication mean. Such elements can be the <methods> element, any of the different media type specific elements as defined in draft-ietf-simple-prescaps-ext [25].

d) a <device> element as defined in draft-ietf-simple-presence-data-model [44] means information about a device.

Additional extensions can be used to express application specific attributes, but their usage is outside the scope of this version of the specification.

5.3.2.3 Subscription for presence information state changes of presentity collections

When the watcher intends to subscribe for presence information state changes of a presentity collection, it shall generate a SUBSCRIBE request in accordance with draft-ietf-simple-event-list [22], additionally to the procedures described in subclause 5.3.2.2.

5.3.2.4 Subscription for the watcher information event template package

Upon activation of the presence service, the watcher may subscribe recursively for the watcher information state changes in accordance with RFC 3857 [28] and RFC 3858 [29].

The watcher may include filters in the body of the SUBSCRIBE request in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

5.3.2.5 Subscription for notification of state changes in XML document

In order to get notifications of changes to XML documents manipulated via the Ut reference point the watcher may generate a SUBSCRIBE request in accordance with draft-ietf-simple-xcap-diff [39] and draft-ietf-sipping-config-framework [43].

5.3.3 Presence Server (PS)

5.3.3.1 General

A PS is an entity that accepts, stores, and distributes presence information.

In addition to the procedures specified in subclause 5.3.3, the PS shall support the procedures specified in 3GPP TS 24.229 [9] appropriate to the functional entity in which the PS is implemented.

5.3.3.2 Subscription acceptance to presence information and notification of state changes

When the PS receives a SUBSCRIBE request for the presence information event package, the PS shall first attempt to verify the identity of the source of the SUBSCRIBE request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then perform authorization according to 3GPP TS 24.229 [9] subclause 5.7.1.5. In case of successful subscription, the PS shall generate a response to the SUBSCRIBE request and notifications in accordance with RFC 3265 [19] and RFC 3586 [27].

Additionally, in the special case of a watcher subscription if the subscription authorization policy results in the action to confirm the watcher subscription from the PUA and the PUA has a valid watcher information subscription, see RFC 3857 [28], then, the PS shall inform the PUA about the watcher subscription attempt.

If the watcher has indicated the need for partial notification using the Accept header field, then the PS shall generate partial notifications in accordance with draft-ietf-simple-partial-notify [24] and draft-ietf-simple-partial-pidf-format [38].

If the body of the SUBSCRIBE request from the watcher contains filters, the PS shall apply the requested filtering function on notifications in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

If the watcher has indicated support for the "multipart/related" content type using the Accept header field, then the PS may generate notifications using "multipart/related" content type which aggregates "application/pidf+xml" formatted presence information with other MIME objects in accordance with RFC 2387 [14]. In this case, the PS shall modify the value of the presence attribute in the PIDF document to refer to the MIME object included in the corresponding MIME multipart body. If the watcher has not indicated support for the "multipart/related" or a MIME object cannot be accessed by the PS, the PS should exclude the presence attribute from the notification.

Editor's note: The details of processing the SUBSCRIBE request and generating the presence document for the watcher are for further study. This material was originally contained within draft-ietf-simple-presence-data-model-01, which was referenced by this specification, but was removed from a later version of the draft. Current work is contained in draft-rosenberg-simple-presence-processing-model.

5.3.3.3 Publication acceptance of presence information

The PS shall act as an Event State Compositor (ESC).

When the PS receives a PUBLISH request, the PS shall first verify the identity of the source of the PUBLISH request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then perform authorization according to 3GPP TS 24.229 [9] subclause 5.7.1.5. In case of successful authentication and authorization, the PS shall process the PUBLISH request in accordance with RFC 3903 [23].

If the PUBLISH request contains the "application/pidf-diff+xml" content-type as described in draft-ietf-simple-partial-pidf-format [38], the PS shall process the PUBLISH request in accordance with RFC 3903 [23] and draft-ietf-simple-partial-publish [45].

If the PUBLISH request contains the "multipart/related" content type and the PS supports the content type, the PS shall process the content as follows:

- if a MIME multipart contains a MIME object of a content type supported by the PS, either store the MIME object in case of initial publication or replace an existing content in case of modify operation; and
- if a multipart includes the "message/external-body" content type and the content indirection is supported by the PS, ensure that it has access to the MIME object indicated by the URI and that the MIME object exists; and

associate the value of the presence attribute that refers to the MIME object with the MIME object and additional information about it.

If the PS does not support the content type used for publishing MIME objects then the PS shall send a 415 (Unsupported Media Type) response and indicate the supported content types in the Accept header.

NOTE: If the PS receives a HTTP request for storing a MIME object on the PS ,meaning that the HTTP URI points to a predefined directory reserved for storing MIME objects and the request is an HTTP PUT request, the PS replaces any existing content referenced by the Request-URI with the content of the request. If the Request-URI points to an uncreated directory, the PS creates the directory, stores the content there and associates the content with the Request-URI. For all requests, i.e. HTTP PUT, HTTP GET and HTTP DELETE requests, the PS generates an appropriate response in accordance with RFC 2616 [15].

5.3.3.4 Subscription acceptance to watcher information and notification of state changes

When the PS receives a SUBCRIBE request for the watcher information event template package, the PS shall first verify the identity of the source of the SUBSCRIBE request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then perform authorization according to 3GPP TS 24.229 [9] subclause 5.7.1.5. In case of successful subscription, the PS shall generate a response to the SUBSCRIBE request and notifications in accordance with RFC 3265 [19], RFC 3857 [28] and RFC 3858 [29].

If the body of the SUBSCRIBE request from the PUA contains filters, the PS shall apply the requested filtering function on notifications in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

5.3.3.5 Subscription acceptance and notification of state changes in XML document

When the PS receives a SUBSCRIBE request having the Event header value "ua-profile", the PS shall first verify the identity of the source of the SUBSCRIBE request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then it shall perform authorization as described in 3GPP TS 24.229 [9] subclause 5.7.1.5. Afterwards, the PS shall generate a response to the SUBSCRIBE request and notifications in accordance with draft-ietf-simple-xcap-diff [39] and draft-ietf-sipping-config-framework [43].

5.3.4 Resource List Server (RLS)

5.3.4.1 General

The Resource List Server (RLS) is an implementation of the presence list server. The RLS is an entity that accepts subscriptions to resource lists and sends notifications to update subscribers of the state of the resources in a resource list.

In addition to the procedures specified in subclause 5.3.4, the RLS shall support the procedures specified in 3GPP TS 24.229 [9] appropriate for an AS in which the RLS is implemented.

5.3.4.2 Subscription acceptance to resource lists and notification of state changes

When the RLS receives a SUBSCRIBE request for the presence information event package of a presentity collection, the RLS shall first verify the identity of the source of the SUBSCRIBE request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then perform authorization according to 3GPP TS 24.229 [9] subclause 5.7.1.5. In case of successful subscription, the RLS shall generate a response to the SUBSCRIBE request and notifications in accordance with draft-ietf-simple-event-list [22] by adding a Require header field with value "eventlist".

If the body of the SUBSCRIBE request from the watcher contains filters, the RLS shall apply the requested filtering function on notifications in accordance with draft-ietf-simple-filter-format [30] and draft-ietf-simple-event-filter-funct [31].

5.3.4.3 Subscription to presence information

When the RLS receives a SUBSCRIBE request for the presence information event package of a presentity collection and installs the corresponding subscription, the RLS shall resolve the list URI to individual URIs and generate SUBSCRIBE requests for each of the individual URIs as per the procedures in RFC 3265 [19], RFC 3856 [27] and draft-ietf-simple-event-list [22] if the state information for the resource represented by the individual URI is otherwise not available.

For internal virtual subscriptions, the detection of loops potentially caused by lists of lists is possible in RLS. However for back-end subscriptions (see draft-ietf-simple-event-list [22]), the detection of such situations is not possible in RLS. To prevent loops in subscriptions to non-local resources the RLS shall not insert "eventlist" in the "Supported" header of back-end subscriptions.

5.3.4.4 Subscription acceptance and notification of state changes in XML document

When the RLS receives a SUBSCRIBE request having the Event header value "ua-profile", the RLS shall first verify the identity of the source of the SUBSCRIBE request as described in 3GPP TS 24.229 [9] subclause 5.7.1.4, then it shall perform authorization as described in 3GPP TS 24.229 [9] subclause 5.7.1.5. Afterwards, the RLS shall generate a response to the SUBSCRIBE request and notifications in accordance with draft-ietf-simple-xcap-diff [39] and draft-ietf-sipping-config-framework [43].

5.3.5 Presence Network Agent (PNA)

5.3.5.1 General

In addition to the procedures specified in subclause 5.3.5, the PNA shall support the procedures specified in 3GPP TS 24.229 [9] appropriate to the functional entity in which the PNA is implemented.

The PNA can collect presence information about the presentity from a number of core network entities. The PNA can combine information from various core network entities to form more complete presence information.

Among these core network entities, the S-CSCF uses SIP to deliver presence information to the PNA over the Pi reference point as specified in subclause 5.3.5.2.

NOTE: As part of the configuration of AS to provide a presence system, appropriate settings are downloaded to the initial filter criteria in the S-CSCF to ensure this occurs. The PNA will receive third-party REGISTER requests as specified in 3GPP TS 24.229 [9] subclauses 5.4.1.7 and 5.7.1.1.

5.3.5.2 Subscription to reg event package

On receiving a third-party REGISTER request which contains an Expires header with a non-zero value, the PNA shall, if no subscription already exists, subscribe to the reg event package for a particular user at the S-CSCF, as described in 3GPP TS 24.229 [9] subclause 5.7.1.1. As a result, the S-CSCF will then provide the presence-related information as reg event packages in NOTIFY requests to the PNA.

On receiving a third-party REGISTER request, the PNA may, if a subscription already exists, resubscribe to the reg event package for a particular user at the S-CSCF, as described in 3GPP TS 24.229 [9] subclause 5.7.1.1. As a result, the S-CSCF will then provide the presence-related information as reg event packages in NOTIFY requests to the PNA.

6 Protocol for data manipulation at the Ut reference point

6.1 Introduction

XML Configuration Access Protocol (XCAP) is used to store, alter and delete data related to the presence service. XCAP is designed according to the Hypertext Transfer Protocol (HTTP) framework, and uses the HTTP methods PUT, GET and DELETE for communication over the Ut reference point. The general information that can be manipulated is user groups, subscription authorization policy, resource lists, hard state presence publication, MIME objects referenced from the hard state presence information, etc. Soft state presence information manipulated with a PUBLISH request is not manipulated by the mechanism provided over the Ut reference point.

6.2 Functional entities

6.2.1 User Equipment (UE)

The UE implements the XCAP client role as described in subclause 6.3.1.

The UE shall implement HTTP digest AKA (see RFC 3310 [20]) and it shall initiate a bootstrapping procedure with the bootstrapping server function located in the home network, as described in 3GPP TS 24.109 [7].

The UE shall acquire the subscriber's certificate from PKI portal by using a bootstrapping procedure, as described in 3GPP TS 24.109 [7].

The UE shall implement HTTP digest authentication (see RFC 2617 [15A]).

The UE shall implement Transport Layer Security (TLS) (see RFC 2246 [13]). The UE shall be able to authenticate the network application function based on the received certificate during TLS handshaking phase.

6.2.2 Application Server (AS)

If an AS implements the role of a PS (see subclause 5.3.3) or of a RLS (see subclause 5.3.4), then the AS shall also implement the role of a XCAP server (see subclause 6.3.2).

If there is no authentication proxy in the network, then the AS shall:

1) implement the role of a network application function, as described in 3GPP TS 24.109 [7];

2) implement TLS (see RFC 2246 [13]);

implement HTTP digest authentication (see RFC 2617 [15A]); and

4) support certificate authentication.

6.2.3 Authentication proxy

The generic requirements for an authentication proxy are defined in 3GPP TS 24.109 [7].

In addition an authentication proxy acting within the scope of presence shall:

1) verify the content of the "X-3GPP-Intended-Identity" header in case it is available in HTTP requests; and

2) indicate an asserted identity of the user in the "X-3GPP-Asserted-Identity" header in HTTP requests sent to the AS.

6.3 Roles

6.3.1 XCAP client

6.3.1.1 Introduction

The XCAP client is a logical function as defined in draft-ietf-simple-xcap [33]. The XCAP client provides the means to manipulate the data such as user groups, subscription authorization policy, resource lists, hard state presence infromation, MIME objects referenced from the hard state presence information, etc.

NOTE: In order to be able to manipulate data stored on the XCAP server, the XCAP client has the root directory on the XCAP server pre-configured or uses some means to discover it. Discovery mechanisms are outside the scope of the present document.

6.3.1.2 Manipulating a resource list

When the XCAP client intends to manipulate a resource list, it shall generate an HTTP PUT, HTTP GET or HTTP DELETE request in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-xcap-list-usage [36].

6.3.1.3 Manipulating the subscription authorization policy

When the XCAP server intends to manipulate the subscription authorization policy, it shall generate an HTTP PUT, HTTP GET or HTTP DELETE request in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-presence-rules [35].

The XCAP client may use an HTTP GET in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-common-policy-caps [41] for fetching of the authorization policy capabilities which the XCAP server supports.

When the XCAP client intends to authorize a different value of the same presence attribute to different watchers or watcher groups, the XCAP client shall authorize a single tuple or person element including one of the different values of the same presence attribute to every watcher or watcher groups as specified in draft-ietf-simple-presence-rules [35].

6.3.1.4 Publishing hard state presence information

The XCAP client shall implement draft-ietf-simple-xcap-pidf-manipulation-usage [34] in order to be able to manipulate hard state presence information. Hard state presence information uses the same format as soft state information, namely "application/pidf+xml" content type as described in RFC 3863 [21] together with any of its extensions.

When the hard state presence information contains one or more MIME objects to be aggregated with the "application/pidf+xml" content type and any of its extensions, the XCAP client shall:

a) construct as many HTTP URIs as the number of objects to be stored and formulate every HTTP URI according to a predefined directory structure;

NOTE: In order to be able to manipulate data stored on the XCAP server, the XCAP client has the root directory on the XCAP server pre-configured or use some means to discover it. Discovery mechanisms are outside the scope of the present document.

b) store the objects on the XCAP server behind the HTTP URI(s) created in the previous step using standard HTTP procedures as defined in RFC 2616 [15];

c) include every HTTP URI as a value of the corresponding XML element in the published "application/pidf+xml" presence document referencing the stored object(s) in the previous step; and

d) publish the hard state presence information according to draft-ietf-simple-xcap-pidf-manipulation-usage [34].

6.3.2 XCAP server

6.3.2.1 Introduction

The XCAP server is a logical function as defined in draft-ietf-simple-xcap [33]. The XCAP server can store data such as user groups, subscription authorization policy, resource lists, hard state presence information, MIME objects referenced from the hard state presence information, etc.

6.3.2.2 Resource list manipulation acceptance

When the XCAP server receives an HTTP PUT, HTTP GET or HTTP DELETE request for manipulating or fetching a resource list, the XCAP server shall first authenticate the request in accordance with 3GPP TS 24.109 [7] and then perform authorization. Afterwards the XCAP server shall perform the requested action and generate a response in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-xcap-list-usage [36].

6.3.2.3 Subscription authorization policy manipulation acceptance

When the XCAP server receives an HTTP PUT, HTTP GET or HTTP DELETE request for manipulating or fetching of the subscription authorization policy, the XCAP server shall first authenticate the request in accordance with 3GPP TS 24.109 [7] and then perform authorization. Afterwards the XCAP server shall perform the requested action and generate a response in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-presence-rules [35].

When the XCAP server receives an HTTP GET request for fetching of the authorization policy capabilities information, the XCAP server shall generate a response in accordance with RFC 2616 [15], draft-ietf-simple-xcap [33] and draft-ietf-simple-pres-policy-caps [42].

6.3.2.4 Publication acceptance of hard state presence information

When the XCAP server receives an HTTP PUT, HTTP GET or HTTP DELETE request for publishing, fetching or deleting of hard state presence information, the XCAP server shall first authenticate the request in accordance with 3GPP TS 24.109 [7] and then perform authorization. Afterwards the XCAP server shall:

a) if the HTTP URI points to a predefined directory reserved for storing MIME objects and the request is an HTTP PUT request, replace any existing content referenced by the Request-URI with the content of the request;

b) if the Request-URI points to an uncreated directory and the request is HTTP PUT, create the directory, store the content there and associate the content with the Request-URI. For all requests, i.e. HTTP PUT, HTTP GET and HTTP DELETE requests, generate an appropriate response in accordance with RFC 2616 [15]; or

c) if the HTTP URI points to an XCAP directory and the Application Unique ID (AUID) part of the HTTP URI is set to "pidf-manipulation", process the request and generate an appropriate response in accordance with draft-ietf-simple-xcap [33], draft-ietf-simple-xcap-pidf-manipulation-usage [34] and RFC 2616 [15].

7 Presence information model of the 3GPP subscriber

7.1 General

Void.

7.2 XML schema definitions

Void.

7.3 XML schema descriptions

Void.

Annex A (informative): Example signalling flows of presence service operation

A.1 Scope of signalling flows

This annex gives examples of signalling flows for the presence service within the IP Multimedia (IM) Core Network (CN) subsystem based on the Session Initiation Protocol (SIP) and SIP Events.

These signalling flows provide detailed signalling flows, which expand on the overview information flows provided in 3GPP TS 23.141 [4].

A.2 Introduction

A.2.1 General

The signalling flows provided in this annex follow the methodology developed in 3GPP TS 24.228 [8]. The following additional considerations apply:

a) 3GPP TS 24.228 [8] shows separate signalling flows with no configuration hiding between networks, and with configuration hiding between networks. There is no presence specific functionality associated with this hiding, and therefore such separate signalling flows are not show in the present document; and

b) 3GPP TS 24.228 [8] does not show the functionality between the S-CSCF and the AS. As the presence service depends on the functionality provided by various AS, the signalling flows between S-CSCF and AS are shown in the present document.

A.2.2 Key required to interpret signalling flows

The key to interpret signalling flows specified in 3GPP TS 24.228 [8] subclauses 4.1 and 4.2 applies with the additions specified below.

- rls.home1.net: an RLS in the home network of the watcher;
- rls.home2.net: an RLS in the home network of the service provider, but not the home network of the watcher;
- ps.home1.net: a PS in the home network of the publisher;
- ps.home2.net: a PS in the home network of the service provider, but not in that of the watcher;
- user1_list1@home1.net: a resource list being subscribed to on a RLS in the home network;
- user2_list1@home2.net: a resource list being subscribed to on a RLS in the home network of the service provider, but not the home network of the subscriber;
- user1_public1@home1.net: presentity being watched, own watcher list;
- user3_public1@home3.net: presentity being watched.

As in 3GPP TS 24.228 [8], in order to differentiate between SIP methods and other protocol messages, the message name is preceded with the associated protocol for all non-SIP messages. Where the XCAP is used to map an HTTP URI to an XML document, the protocol name "XCAP" is used for both the HTTP request and HTTP response.

Each signalling flow table contains descriptions for headers where the content of the header is new to that signalling flow, as is already performed in 3GPP TS 24.228 [8].

However, 3GPP TS 24.228 [8] includes extensive descriptions for the contents of various headers following each of the tables representing the contents of the signalling flows. Where the operation of the header is identical to that shown in 3GPP TS 24.228 [8], then such text is not reproduced in the present document.

Additional text may also be found on the contents of headers within 3GPP TS 24.228 [8] in addition to the material shown in the present document.

A.3 Signalling flows demonstrating how watchers subscribe to presence event notification

A.3.1 Introduction

The subclause covers the signalling flows that show how watchers can request presence information about a presentity.

For the routing of the Public Service Identity (PSI) towards the AS, there are two scenarios:

Subclause A.3.3.2 shows the case where the I-CSCF forwards the SUBSCRIBE request directly to the RLS when the RLS is located within the same network. There is another scenario where the I-CSCF forwards the SUBSCRIBE request towards the RLS, being involved with the S-CSCF located in the same network, but this scenario is not described in the present document.

A.3.2 Watcher and presentity in different networks, UE in home network

A.3.2.1 Successful subscription

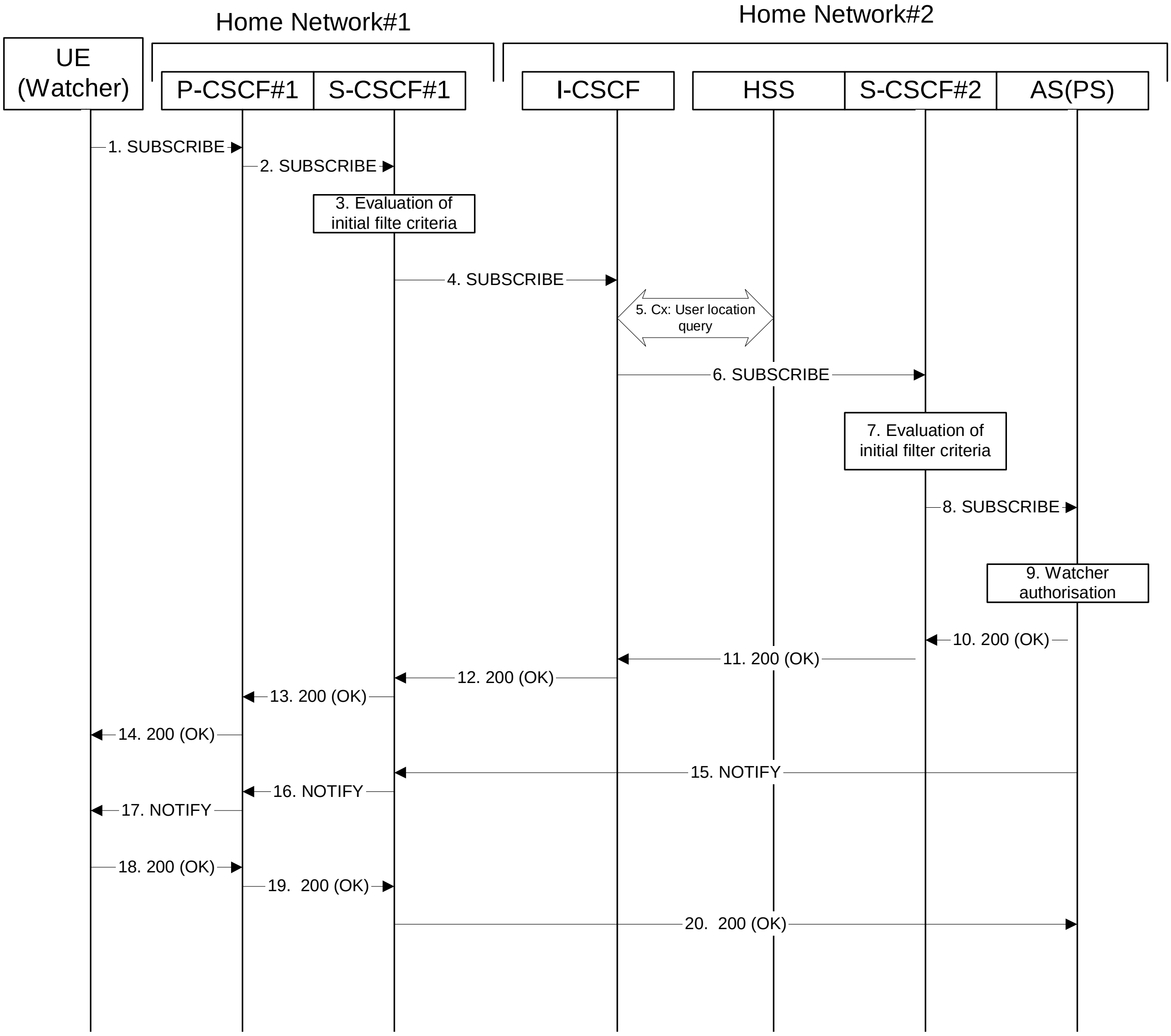

Figure A.3.2.1-1: Watcher subscribing for presence information

Figure A.3.2.1-1 shows a watcher subscribing to presence event notification about a presentity. The presentity is in a different IM CN subsystem. The details of the signalling flows are as follows:

1. **SUBSCRIBE request (UE (watcher) to P-CSCF) - see example in table A.3.2.1-1**

   A watcher agent in a UE wishes to watch a presentity, or certain presence information of the presentity. To initiate a subscription, the UE generates a SUBSCRIBE request containing the "presence" event that it wishes to be notified of, together with an indication of the length of time this periodic subscription should last and the support for partial notification.

Table A.3.2.1-1: SUBSCRIBE request (UE (watcher) to P-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user2_public1@home2.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 61 SUBSCRIBE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: presence
Expires: 7200
Accept: application/pidf+xml;q=0.3, application/pidf-diff+xml;q=1
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request-URI: Public user identity whose events the subscriber subscribes to.

Event: This field is populated with the value "presence" to specify the use of the presence package.

Accept: This field is populated with the value 'application/pidf+xml' and 'application/pidf-diff+xml', latter one with higher preference.

To: Same as the Request-URI.

2. **SUBSCRIBE request (P-CSCF to S-CSCF) - see example in table A.3.2.1-2**

The P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The SUBSCRIBE request is forwarded to S-CSCF. A Route header is inserted into SUBSCRIBE request. The information for the Route header is taken from the service route determined during registration.

Table A.3.2.1-2: SUBSCRIBE request (P-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Access-Network-Info:
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
Route: <sip:orig@scscf1.home1.net;lr>
Record-Route: <sip:pcscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

3. **Evaluation of initial filter criteria**

S-CSCF#1 validates the service profile of this subscriber and evaluates the initial filter criteria. In this example, no AS is assumed to be involved.

4. **SUBSCRIBE request (S-CSCF to I-CSCF) - see example in table A.3.2.1-4**

S-CSCF#1 performs an analysis of the destination address, and determines the network operator to whom the destination subscriber belongs. Since the originating operator does not desire to keep their internal configuration hidden, S-CSCF#1 forwards the SUBSCRIBE request directly to the I-CSCF in the destination network.

Table A.3.2.1-4: SUBSCRIBE (S-CSCF to I-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;;branch=z9hG4bKnashds7
Max-Forwards: 68
P-Asserted-Identity: <sip:user1_public1@home1.net>, <tel:+1-212-555-1111>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
Privacy:
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

5. **Cx: User Location Query procedure**

The I-CSCF sends a query to the HSS to find out the S-CSCF of the called user. The HSS responds with the address of the current S-CSCF for the terminating subscriber.

For detailed message flows see 3GPP TS 29.228 [10].

Table A.3.2.1-5a provides the parameters in the SIP SUBSCRIBE request (flow 4), which are sent to the HSS.

Table A.3.2.1-5a: Cx: User registration status query procedure (I-CSCF to HSS)

Message source and destination	Cx: Information element name	Information source in SIP SUBSCRIBE	Description
I-CSCF to HSS	User Public Identity	Request-URI	This information element indicates the public user identity

Table A.3.2.1-5b provides the parameters sent from the HSS that need to be mapped to the SIP SUBSCRIBE request (flow 6) and sent to the S-CSCF.

Table A.3.2.1-5b: Cx: User registration status query procedure (HSS to I-CSCF)

Message source and destination	Cx: Information element name	Mapping to SIP header in SIP SUBSCRIBE	Description
HSS to I-CSCF	S-CSCF name	Route header field	This information indicates the serving CSCF's name of that user

6. **SUBSCRIBE request (I-CSCF to S-CSCF) - see example in table A.3.2.1-6**

   The I-CSCF forwards the SUBSCRIBE request to the S-CSCF (S-CSCF#2) that will handle the termination.

Table A.3.2.1-6: SUBSCRIBE request (I-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 67
P-Asserted-Identity:
P-Charging-Vector:
Privacy:
Route: <sip:scscf2.home2.net;lr>
Record-Route:
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

NOTE: The I-CSCF does not add itself to the Record-Route header, as it has no need to remain in the signalling path for the subsequent requests.

7. **Evaluation of initial filter criteria**

   S-CSCF#2 validates the service profile of this subscriber and evaluates the initial filter criteria. For sip:user2_public1@home2.net S-CSCF#2 has termination initial filter criteria with Service Point Trigger of Method = SUBSCRIBE and Event = "presence" that informs the S-CSCF to route the SUBSCRIBE request to the AS ps.home2.net. The S-CSCF#2 has preconfigured information not to create a Record-Route entry for this request.

8. **SUBSCRIBE request (S-CSCF to PS) – see example in table A.3.2.1-8**

   The S-CSCF forwards the SUBSCRIBE request to the PS.

Table A.3.2.1-8: SUBSCRIBE request (S-CSCF to PS)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
      icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 66
P-Asserted-Identity:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Route: <sip:ps.home2.net;lr>, <sip:scscf2.home2.net;lr>
Record-Route:
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the originating

Inter Operator Identifier (IOI) parameter of this header and removes the terminating IOI parameter.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the PS.

9. **Authorization of watcher**

The PS performs the necessary authorization checks on the originator to ensure it is allowed to watch the presentity. In this example all privacy conditions are met, so the PS sends a 200 (OK) response to the S-CSCF.

In the case where the privacy/authorization checks failes, then a necessary 2xx or 4xx response will be sent to the S-CSCF. The selection of the correct response code depends on the presentity's subscription authorization policy document.

10. **200 (OK) response (PS to S-CSCF) - see example in table A.3.2.1-10**

The PS sends the response to S-CSCF#2.

Table A.3.2.1-10: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net;
      term-ioi=home2.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route:
From:
To: <sip:user2_public1@home2.net>;tag=151170
Call-ID:
CSeq:
Expires:
Contact: <sip:ps.home2.net>
Content-Length:
```

P-Charging-Vector: The PS stores the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS stores the P-Charging-Function-Addresses header field and passes this header to the S-CSCF.

11. **200 (OK) response (S-CSCF to I-CSCF) - see example in table A.3.2.1-11**

S-CSCF#2 forwards the response to I-CSCF#2.

Table A.3.2.1-11: 200 (OK) response (S-CSCF to I-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector:
P-Charging-Function-Addresses:
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the I-CSCF.

12. **200 (OK) response (I-CSCF to S-CSCF) - see example in table A.3.2.1-12**

I-CSCF#2 forwards the response to S-CSCF#1.

Table A.3.2.1-12: 200 (OK) response (I-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector:
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

13. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.3.2.1-13**

S-CSCF#1 forwards the response to P-CSCF#1.

Table A.3.2.1-13: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

14. **200 (OK) response (P-CSCF to UE) - see example in table A.3.2.1-14**

P-CSCF#1 forwards the response to the watcher agent in the UE.

Table A.3.2.1-14: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

15. **NOTIFY request (PS to S-CSCF) - see example in table A.3.2.1-15**

As soon as the PS sends a 200 (OK) response to accept the subscription, it sends a NOTIFY request with the current state of the presentity's presence information that the watcher has subscribed and been authorized to.

The NOTIFY request is sent to S-CSCF#1. Based on the Accept header field of the SUBSCRIBE request, the PS decides to use partial notification to provide changes of presence information.

Table A.3.2.1-15: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"; orig-ioi=home2.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net;lr>
From: <sip:user2_public1@home2.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 42 NOTIFY
Subscription-State: active ;expires=7200
Event: presence
Contact: <sip:ps.home2.net>
Content-Type: application/pidf-diff +xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   <diff:pidf-full xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns:diff="urn:ietf:params:xml:ns:pidf-diff"

             xmlns: rp="urn:ietf:params:xml:ns:pidf:rpid"

             xmlns:dm="urn:ietf:params:xml:ns:pidf: data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
        entity="pres:user2_public1@home2.net"
        version="1">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>sip</rp:class>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>assistant</rp:class>
       <rp:relationship><rp:assistant/></rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person id="s438">
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>
         <rp:activities><rp:meeting/></rp:activities>
         <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>
     </dm:person>

   </diff:pidf-full>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique identifier and adds the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

Content-Type: Set to the preferred value of the Accept header received in the SUBSCRIBE request.

The message body in the NOTIFY request that carries the presence information of the presentity is formed as indicated in RFC 3863 [21], draft-ietf-simple-rpid [26], draft-ietf-simple-prescaps-ext [25], draft-ietf-simple-cipid [32] draft-ietf-simple-partial-notify [24] and draft-ietf-simple-presence-data-model [44].

16. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.2.1-16**

The S-CSCF#1 forwards the NOTIFY request to P-CSCF#1.

Table A.3.2.1-16: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"
P-Charging-Function-Addresses:
Privacy:
Record-Route: <sip:scscf1.home1.net;lr>
Route: sip:<pcscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter of this header and removes the parameter from this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

17. **NOTIFY request (P-CSCF to UE) - see example in table A.3.2.1-17**

The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.2.1-17: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
Privacy:
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

18. **200 (OK) response (UE to P-CSCF) - see example in table A.3.2.1-18**

The UE generates a 200 (OK) response to the NOTIFY request.

Table A.3.2.1-18: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

19. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.2.1-19**

The P-CSCF forwards the 200 (OK) response to S-CSCF#1.

Table A.3.2.1-19: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info:
From:
To:
Call-ID:
CSeq:
Content-Length:
```

20. **200 (OK) response (S-CSCF to P-S) - see example in table A.3.2.1-20**

S-CSCF#2 forwards the 200 (OK) response to the PS.

Table A.3.2.1-20: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
From:
To:
Call-ID:
CSeq:
Content-Length:
```

A.3.3 Watcher subscribing to resource list, UE in visited network

A.3.3.1 Watcher subscribing to his own resource list, UE in visited network - Successful subscription

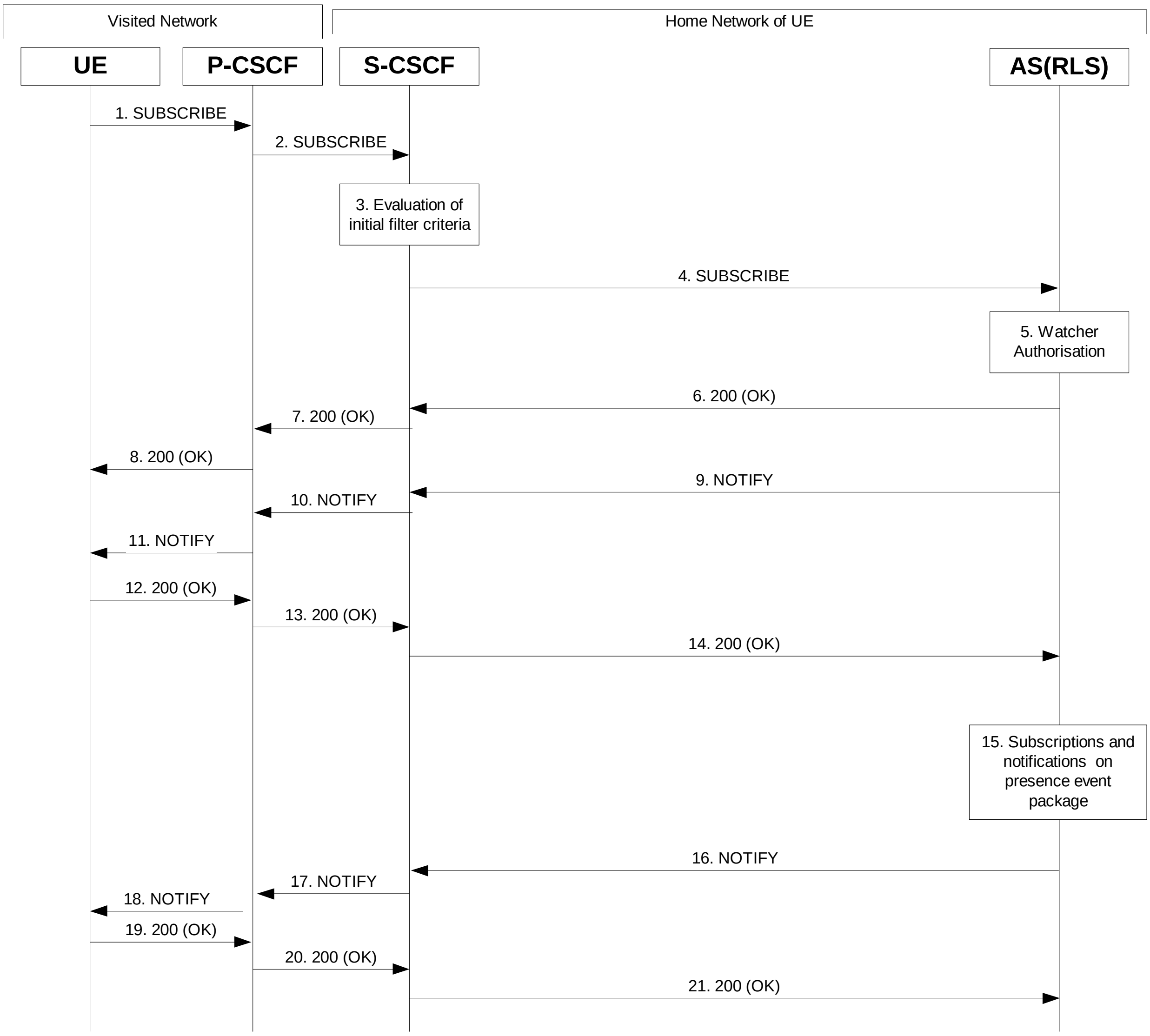

Figure A.3.3.1-1: Watcher subscribing to resource list

Figure A.3.3.1-1 shows a watcher subscribing to resource list event notification. The details of the signalling flows are as follows:

1. **SUBSCRIBE request (UE to P-CSCF) – see example in table A.3.3.1-1**

   A watcher agent in a UE wishes to watch a number of presentities, or certain presence information of these presentities. The list of presentities are identified by a SIP URI. In order to initiate a subscription to the RLS, the UE generates a SUBSCRIBE request indicating support for "eventlist", together with an indication of the length of time this periodic subscription should last.

Table A.3.3.1-1: SUBSCRIBE request (UE to P-CSCF)

```
SUBSCRIBE sip:user1_list1@home1.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user1_list1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 123 SUBSCRIBE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: presence
Supported: eventlist
Expires: 7200
Accept: application/pidf+xml, application/rlmi+xml, multipart/related
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request-URI: SIP URI of the resource list representing the collection of public user identities whose events the subscriber subscribes to.

Event: This field is populated with the value "presence" to specify the use of the presence package.

Accept: This field is populated with the value "application/pidf+xml", "application/rlmi+xml" and "multipart/related" indicating that the UE supports both body types for the eventlist extension additionally to PIDF.

Supported: This field is populated with the value "eventlist" to specify the support for the eventlist extension.

To: Same as the Request-URI.

2. **SUBSCRIBE request (P-CSCF to S-CSCF) – see example in table A.3.3.1-2**

   The P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The SUBSCRIBE request is forwarded to S-CSCF#1. A Route header is inserted into SUBSCRIBE request. The information for the Route header is taken from the service route determined during registration.

Table A.3.3.1-2: SUBSCRIBE request (P-CSCF to S-CSCF)

```
SUBSCRIBE sip:user1_list1@home1.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Access-Network-Info:
Route: <sip:orig@scscf1.home1.net;lr>
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
Privacy:
Record-Route: <sip:pcscf1.visited1.net;lr>
Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

3. **Evaluation of initial filter criteria**

The S-CSCF validates the service profile of this subscriber and evaluates the initial filter criteria. Assuming that sip:user1_list1@home1.net is a statically created PSI, sip:user1_list1@home1.net is included in the service profile as part of an originating initial Filter Criteria with Service Trigger Point of Method = SUBSCRIBE AND Supported = "eventlist" AND Request-URI = sip:user1_list1@home1.net that informs the S-CSCF to route the SUBSCRIBE request to the AS sip:rls.home1.net.

If there is no initial filter criteria for this PSI (sip:user1_list1@home1.net), the assumption is that the PSI is a sub domain-based PSI. The procedure defined in RFC 3263 [18] with DNS NAPTR and SRV queries may then be used to get the IP address of the AS home1.net.

4. **SUBSCRIBE request (S-CSCF to RLS) – see example in table A.3.3.1-4**

The S-CSCF forwards the SUBSCRIBE request to the RLS.

Table A.3.3.1-4: SUBSCRIBE request (S-CSCF to RLS)

```
SUBSCRIBE sip:user1_list1@home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 68
P-Access-Network-Info:
P-Asserted-Identity: <sip:user1_public1@home1.net>, <tel:+1-212-555-1111>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
Route: <sip:rls.home1.net;lr>, <sip:orig@scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the RLS.

5. **Authorization of watcher**

The RLS performs the necessary authorization checks on the originator to ensure that he/she is authorized to use the resource list. In this example this condition has been met, so the PS sends a 200 (OK) response to the S-CSCF. If the previous condition failed, then a 403 (Forbidden) response would be sent to the S-CSCF.

6. **200 (OK) response (RLS to S-CSCF) - see example in table A.3.3.1-6**

The RLS sends the response to the S-CSCF.

Table A.3.3.1-6: 200 (OK) response (RLS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net;
      term-ioi=home1.net
Record-Route:
From:
To: <sip:user1_list1@home1.net>;tag=151170
Call-ID:
CSeq:
Require: eventlist
Expires:
Contact:
Content-Length: 0
```

P-Charging-Vector: The RLS stores the terminating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

7. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.3.3.1-7**

   The S-CSCF forwards the response to the P-CSCF.

Table A.3.3.1-7: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
Record-Route:
From:
To:
Call-ID:
CSeq:
Require:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter received.

8. **200 (OK) response (P-CSCF to UE) - see example in table A.3.3.1-8**

   The P-CSCF forwards the response to the watcher agent in the UE.

Table A.3.3.1-8: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Require:
Expires:
Contact:
Content-Length:
```

9. **NOTIFY request (RLS to S-CSCF) - see example in table A.3.3.1-9**

   The RLS generates a NOTIFY request including the RLMI document as a result of the SUBSCRIBE request.

Table A.3.3.1-9 NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_list1@home1.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 89 NOTIFY
Subscription-State: active;expires=7200
Require: eventlist
Event: presence
Contact: <sip:rls.home1.net>
Content-Type: application/rlmi+xml;charset="UTF-8"
Content-Length:

<?xml version="1.0" encoding="UTF-8"?>
  <list xmlns="urn:ietf:params:xml:ns:rmli"
       uri="sip:user1_list1@home1.net" version="1" fullState="true">
     <resource uri="pres:user2_public1@home2.net">
       <name>Kovacs Janos</name>
       <instance id="hqzsuxtfyq" state="active" cid="ZvSvkz@rls.home1.net"/>
     </resource>
     <resource uri="pres:user3_public1@home3.net">
       <name>Szabo Bela</name>
       <instance id="aakdsjklsa" state="active" cid="HJjbssk@rls.home1.net"/>
     </resource>
  </list>
```

P-Charging-Vector: The RLS inserts this header and populates the icid parameters with a globally unique value and adds the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The RLS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

10. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.3.1-10**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.3.3.1-10: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores originating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

11. **NOTIFY request (P-CSCF to UE) – see example in table A.3.3.1-11**

    The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.3.1-11: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Length:

(…)
```

12. **200 (OK) response (UE to P-CSCF) – see example in table A.3.3.1-12**

    The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.3.3.1-12: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

13. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.3.1-13**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.3.3.1-13: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

14. **200 (OK) response (S-CSCF to RLS) - see example in table A.3.3.1-14**

    The S-CSCF#2 forwards the response to the RLS in the home network of the UE.

Table A.3.3.1-14: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net;
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

15. **Subscriptions and notifications on presence event package**

After the RLS generated a NOTIFY request to inform the UE about the subscription state, the RLS generates the necessary SUBSCRIBE requests to the presentities present in the resource list as described in subclause A.3.4.1. As soon as it receives NOTIFY request(s) about a state change in one or more presentities, it generates a NOTIFY request.

16. **NOTIFY request (RLS to S-CSCF) – see example in table A.3.3.1-16**

The RLS copies the body of the incoming NOTIFY request(s) into the body of the outgoing NOTIFY request using MIME type multipart/related. Further notification sent by the RLS may contain either the full or the partial set of presence information (only the presence information that has changed since the last notification) as described in draft-ietf-simple-event-list [22].

In this example it is assumed that the RLS has received two NOTIFY requests from presentities sip:user2_public1@home2.net and sip:user3_public1@home3.net before generating the NOTIFY request in table A.3.3.1-16 to the UE.

Table A.3.3.1-16 NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_list1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 90 NOTIFY
Subscription-State: active;expires=5000
Require: eventlist
Event: presence
Contact: <sip:rls.home1.net>
Content-Type: multipart/related;type="application/rlmi+xml";
      start="<nXYxAE@rls.home1.net>";boundary="50UBfW7LSCVLtggUPe5z"
Content-Length: (...)

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <nXYxAE@rls.home1.net>
Content-Type: application/rlmi+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
  <list xmlns="urn:ietf:params:xml:ns:rmli"
        uri="sip:user1_list1@home1.net" version="1" fullState="true">
     <resource uri="pres:user2_public1@home2.net">
       <name>Kovacs Janos</name>
       <instance id="hqzsuxtfyq" state="active" cid="ZvSvkz@rls.home1.net"/>
     </resource>
     <resource uri="pres:user3_public1@home3.net">
       <name>Szabo Bela</name>
       <instance id="aakdsjklsa" state="active" cid="HJjbssk@rls.home1.net"/>
     </resource>
  </list>

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <ZvSvkz@rls.home1.net>
Content-Type: application/pidf+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns: rp="urn:ietf:params:xml:ns:pidf:rpid"
             xmlns:dm="urn:ietf:params:xml:ns:pidf: data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
             entity="pres:user2_public1@home2.net">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
       </status>
       < rp:class>sip</ rp:class>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       < rp:class>assistant</ rp:class>
       < rp:relationship><rp:assistant/></ rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person>
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>

       <rp:activities><rp:meeting/></rp:activities>
       <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>

     </dm:person>

   </presence>

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <ZvSvkz@pres.example.com>
Content-Type: application/pidf+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns: rp="urn:ietf:params:xml:ns:pidf:rpid"


             xmlns:dm="urn:ietf:params:xml:ns:pidf: data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             entity="pres:user3_public1@home3.net">

     <tuple id="h7833hjkk.dsajfjdsaf">
       <status>
         <basic>closed</basic>
       </status>
       < rp:class>sip</ rp:class>
       <rp:privacy><rp:text/></rp:privacy>
       <pcp:servcaps>
           <pcp:video>false</pcp:video>
           <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user3_public1@home3.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="hu">Senki se merjen zavarni!</note>
       <timestamp>2003-08-27T11:48:59Z</timestamp>
     </tuple>

     <tuple id="sajdhdsahjh75vvcb774.78">
       <status>
         <basic>open</basic>
       </status>
       < rp:class>supervisor</ rp:class>
       < rp:relationship><rp:supervisor/></ rp:relationship>
       <contact priority="1.0">tel:+1-858-204-9141</contact>
       <note xml:lang="en">He's my supervisor</note>
       <timestamp>2003-08-27T11:48:59Z</timestamp>
     </tuple>

     <dm:person>
       <ci:homepage>http://example.com/~user3</ci:homepage>
       <ci:card>http://example.com/~user3/card.vcd</ci:card>
       < rp:class>presentity</ rp:class>
       <rp:activities><ep:vacation/></rp:activities>
       <rp:place-type until="2003-09-10T17:30:00Z"><rp:ship/></rp:place-type>
     </dm:person>

   </presence>
```

```
--50UBfW7LSCVLtggUPe5z--
```

P-Charging-Vector: The RLS inserts this header and populates the icid parameters with a globally unique value and adds the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The RLS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

Content-Type: Set to the value of the Accept: header received in the SUBSCRIBE request.

The message body in the NOTIFY request that carries the presence information of the presentity is formed as indicated in draft-ietf-simple-event-list [22], draft-ietf-simple-presence-data-model [44], draft-ietf-simple-rpid [26], draft-ietf-simple-cipid [32] and draft-ietf-simple-prescaps-ext [25].

17. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.3.1-17**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.3.3.1-17: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content Type:
Content-Length:

(…)
```

P-Charging-Vector: The RLS stores the originating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

18. **NOTIFY request (P-CSCF to UE) - see example in table A.3.3.1-18**

    The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.3.1-18: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Content-Type:
Content-Length:

(…)
```

19. **200 (OK) response (UE to P-CSCF) – see example in table A.3.3.1-19**

    The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.3.3.1-19: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

20. **200 (OK) response (P-CSCF to S-CSCF) - see example in table A.3.3.1-20**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.3.3.1-20: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

21. **200 (OK) response (S-CSCF to RLS) - see example in table A.3.3.1-21**

The S-CSCF#2 forwards the response to the RLS in the home network of the UE.

Table A.3.3.1-21: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home1.net;
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

A.3.3.2 Watcher subscribing to a resource list, UE in visited network - successful subscription

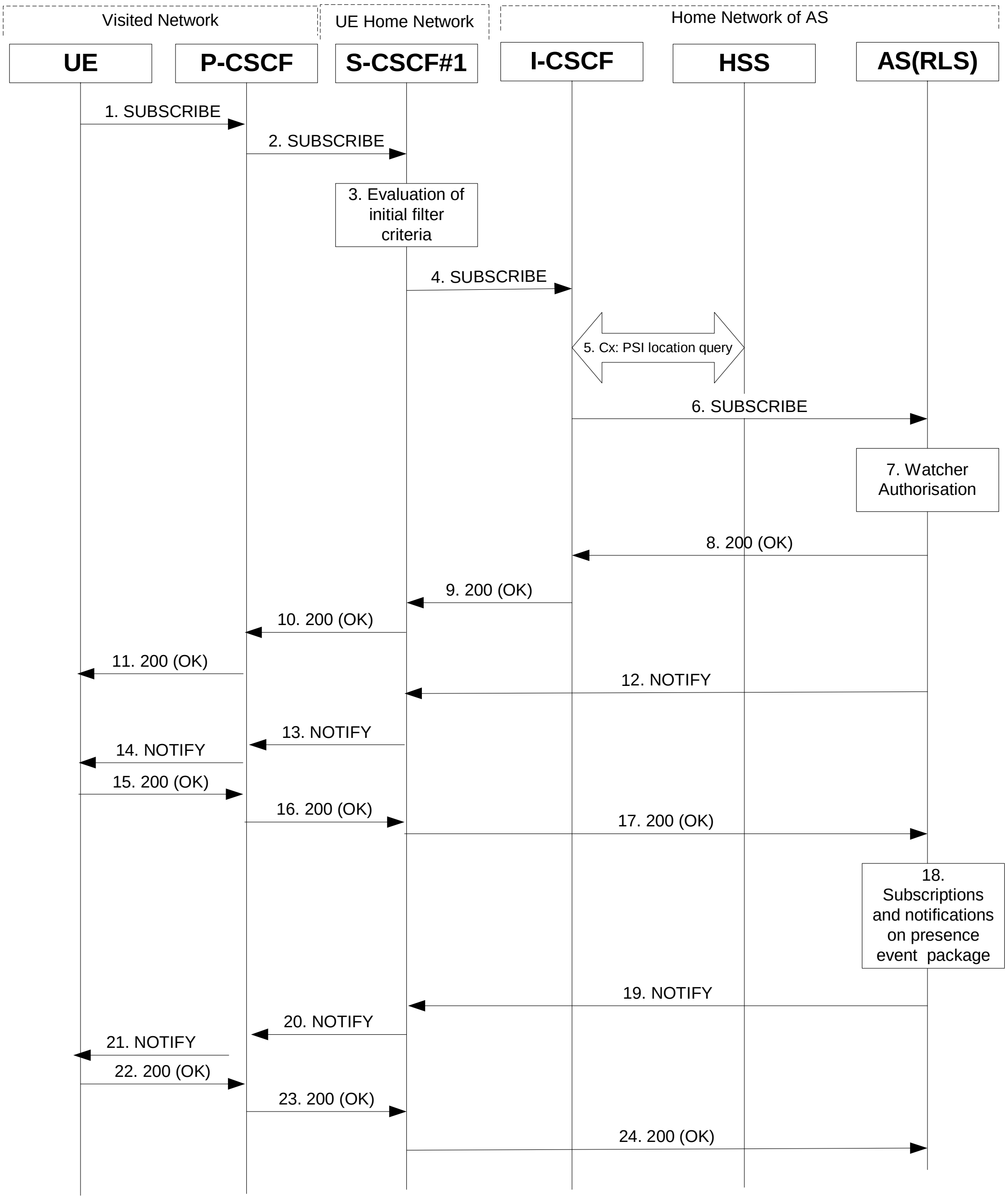

Figure A.3.3.2-1 Watcher subscribing to resource list

Figure A.3.3.2-1 shows a watcher subscribing to resource list event notification. The details of the signalling flows are as follows:

1. **SUBSCRIBE request (UE to P-CSCF) - see example in table A.3.3.2-1**

   A watcher agent in a UE wishes to watch a number of presentities, or certain presence information of these presentities. The list of presentities are identified by a SIP URI. In order to initiate a subscription to the RLS, the UE generates a SUBSCRIBE request indicating support for "eventlist", together with an indication of the length of time this periodic subscription should last.

Table A.3.3.2-1: SUBSCRIBE request (UE to P-CSCF)

```
SUBSCRIBE sip:user2_list1@home2.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user2_list1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 123 SUBSCRIBE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: presence
Supported: eventlist
Expires: 7200
Accept: application/pidf+xml, application/rlmi+xml, multipart/related
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request-URI: SIP URI of the resource list representing the collection of public user identities whose events the subscriber subscribes to.

Event: This field is populated with the value "presence" to specify the use of the presence package.

Accept: This field is populated with the value "application/pidf+xml", "application/rlmi+xml" and "multipart/related" indicating that the UE supports the eventlist extension additionally to PIDF.

Supported: This field is populated with the value "eventlist" to specify the support for the eventlist extension.

To: Same as the Request-URI.

2. **SUBSCRIBE request (P-CSCF to S-CSCF) - see example in table A.3.3.2-2**

   The P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The SUBSCRIBE request is forwarded to S-CSCF#1. A Route header is inserted into SUBSCRIBE request. The information for the Route header is taken from the service route determined during registration.

Table A.3.3.2-2: SUBSCRIBE request (P-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_list1@home2.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Access-Network-Info:
Route: <sip:orig@scscf1.home1.net;lr>
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
Record-Route: <sip:pcscf1.visited1.net;lr>
Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

3. **Evaluation of initial filter criteria**

   S-CSCF#1 validates the service profile of this subscriber and evaluates the initial filter criteria. In this example, no AS is assumed to be involved.

4. **SUBSCRIBE request (S-CSCF to I-CSCF) - see example in table A.3.3.2-4**

   S-CSCF#1 performs an analysis of the destination address. As the destination address points to a resource that is in a different network as the S-CSCF, the S-CSCF sends the request to the I-CSCF of home2.net.

Table A.3.3.2-4: SUBSCRIBE request (S-CSCF to I-CSCF)

```
SUBSCRIBE sip:user2_list1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 68
P-Asserted-Identity: <sip:user1_public1@home1.net>, <tel:+1-212-555-1111>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Record-Route: <orig@sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

5. **PSI location query**

The I-CSCF sends a query to the HSS to find the RLS where sip:user2_list1@home2.net is hosted. The HSS responds with the address of the RLS.

For detailed message flows see 3GPP TS 29.228 [10].

Table A.3.3.2-5a provides the parameters in the SIP SUBSCRIBE request (flow 4), which are sent to the HSS.

Table A.3.3.2-5a Cx: User registration status query procedure (I-CSCF to HSS)

Message source & destination	Cx: Information element name	Information source in SIP SUBSCRIBE	Description
I-CSCF to HSS	User Public Identity	Request-URI:	This information element indicates the PSI of the RLS

Table A.3.3.2-5b provides the parameters sent from the HSS that need to be mapped to SIP SUBSCRIBE (flow 6) and sent to S-CSCF.

Table A.3.3.2-5b Cx: User registration status query procedure (HSS to I-CSCF)

Message source & destination	Cx: Information element name	Mapping to SIP header in SIP SUBSCRIBE	Description
HSS to I-CSCF	S-CSCF name	Route header field	This information indicates the address of the RLS

6. **SUBSCRIBE request (I-CSCF to RLS) - see example in table A.3.3.2-6**

The I-CSCF forwards the SUBSCRIBE request to the RLS.

Table A.3.3.2-6: SUBSCRIBE request (I-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_list1@home2.net SIP/2.0
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 67
P-Asserted-Identity:
P-Charging-Vector:
Privacy:
Record-Route:
Route: <sip:rls.home2.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

7. **Authorization of watcher**

The RLS performs the necessary authorization checks on the originator to ensure that he/she is authorized to use the resource list. In this example this condition has been met, so the PS sends a 200 (OK) response to the S-CSCF. If the previous condition failed, then a 403 (Forbidden) response would be sent to the S-CSCF.

8. **200 (OK) response (RLS to I-CSCF) - see example in table A.3.3.2-8**

The RLS sends the response to the S-CSCF.

Table A.3.3.2-8: 200 (OK) response (RLS to I-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net;
      term-ioi=home2.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route:
From:
To: <sip:user2_list1@home2.net>;tag=151170
Call-ID:
CSeq:
Require: eventlist
Expires:
Contact: <sip:rls.home2.net>
Content-Length: 0
```

P-Charging-Vector: The RLS stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The RLS stores the P-Charging-Function-Addresses header field and passes this header to the I-CSCF.

9. **200 (OK) response (I-CSCF to S-CSCF) - see example in table A.3.3.2-9**

The I-CSCF forwards the response to the S-CSCF.

Table A.3.3.2-9: 200 (OK) response (I-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector:
Record-Route:
From:
To:
Call-ID:
CSeq:
Require:
Expires:
Contact:
Content-Length: 0
```

P-Charging-Vector: The RLS stores the header and passes this header to the S-CSCF.

10. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.3.3.2-10**

    The S-CSCF forwards the response to the P-CSCF.

Table A.3.3.2-10: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Record-Route:
From:
To:
Call-ID:
CSeq:
Require:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter.

11. **200 (OK) response (P-CSCF to UE) - see example in table A.3.3.2-11**

    The P-CSCF forwards the response to the watcher agent in the UE.

Table A.3.3.2-11: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Require:
Expires:
Contact:
Content-Length:
```

12. **NOTIFY request (RLS to S-CSCF) – see example in table A.3.3.2-12**

The RLS generates a NOTIFY request including the RLMI document as a result of the SUBSCRIBE request.

Table A.3.3.2-12: NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"; orig-ioi=home1.net
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user2_list1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 89 NOTIFY
Subscription-State: active;expires=5000
Require: eventlist
Event: presence
Contact: <sip:rls.home1.net>
Content-Type: application/rlmi+xml;charset="UTF-8"
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
  <list xmlns="urn:ietf:params:xml:ns:rmli"
        uri="sip:user1_list1@home1.net" version="1" fullState="true">
     <resource uri="pres:user2_public1@home2.net">
       <name>Kovacs Janos</name>
       <instance id="hqzsuxtfyq" state="active" cid="ZvSvkz@rls.home2.net"/>
     </resource>
     <resource uri="pres:user3_public1@home2.net">
       <name>Szabo Bela</name>
       <instance id="aakdsjklsa" state="active" cid="HJjbssk@rls.home2.net"/>
     </resource>
  </list>
```

P-Charging-Vector: The RLS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

13. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.3.2-13**

The S-CSCF#1 forwards the NOTIFY request to the P-CSCF.

Table A.3.3.2-13: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Max-Forwards: 69
Record-Route: <sip:scscf1.home1.net;lr>
Route: <sip:pcscf1.visited1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter.

P-Charging-Function-Addresses: The RLS populates the P-Charging-Function-Addresses header field to be passed to the I-CSCF.

14. **NOTIFY request (P-CSCF to UE) - see example in table A.3.3.2-14**

    The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.3.2-14: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event: Contact:
Content-Type:
Content-Length:

(…)
```

15. **200 (OK) response (UE to P-CSCF) - see example in table A.3.3.2-15**

    The UE acknowledges the NOTIFY request with a 200 (OK) to the P-CSCF.

Table A.3.3.2-15: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

16. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.3.2-16**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF#1.

Table A.3.3.2-16: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

17. **200 (OK) response (S-CSCF to RLS) - see example in table A.3.3.2-17**

    The S-CSCF#1 forwards the response to the RLS in the home network of the UE.

Table A.3.3.2-17: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"; orig-ioi=home1.net:
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

18. **Subscriptions and notifications on presence event package**

    After the RLS generated a 200 (OK) response to the SUBSCRIBE request from the UE, it generates the necessary SUBSCRIBE requests to the presentities present in the resource list as described in subclause A.3.4.1. As soon as it receives NOTIFY request(s) about a state change in one or more presentities, it generates a NOTIFY request.

19. **NOTIFY request (RLS to S-CSCF) – see example in table A.3.3.2-19**

    The RLS copies the body of the incoming NOTIFY request(s) into the body of the outgoing NOTIFY request using MIME type multipart/related. Further notification sent by the RLS contain may contain either the full or the partial set of presence information (only the presence information that has changed since the last notification) as described in draft-ietf-simple-event-list [22].

    In this example it is assumed that the RLS receives two NOTIFY requests from presentities sip:user2_public1@home2.net and sip:user3_public1@home3.net before generating the NOTIFY request in subclause A.3.3.2-23 to the UE.

Table A.3.3.2-19: NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user2_list1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 89 NOTIFY
Subscription-State: active;expires=5000
Require: eventlist
Event: presence
Contact: <sip:rls.home2.net>
Content-Type: multipart/related;type="application/rlmi+xml";
       start="<nXYxAE@rls.home2.net>";boundary="50UBfW7LSCVLtggUPe5z"
Content-Length: (...)

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <nXYxAE@rls.home2.net>
Content-Type: application/rlmi+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
  <list xmlns="urn:ietf:params:xml:ns:rmli"
        uri="sip:user1_list1@home1.net" version="1" fullState="true">
     <resource uri="pres:user2_public1@home2.net">
       <name>Kovacs Janos</name>
       <instance id="hqzsuxtfyq" state="active" cid="ZvSvkz@rls.home2.net"/>
     </resource>
     <resource uri="pres:user3_public1@home3.net">
       <name>Szabo Bela</name>
       <instance id="aakdsjklsa" state="active" cid="HJjbssk@rls.home2.net"/>
     </resource>
   </list>

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <ZvSvkz@rls.home2.net>
Content-Type: application/pidf+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns: rp="urn:ietf:params:xml:ns:pidf:rpid"


             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidfcaps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
             entity="pres:user2_public1@home2.net">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
       </status>
       < rp:class>sip</ rp:class>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>assistant</rp:class>

       <rp:relationship><rp:assistant/></rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person>
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>
       <rp:activities><rp:meeting/></rp:activities>
       <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>
     </dm:person>

   </presence>

--50UBfW7LSCVLtggUPe5z
Content-Transfer-Encoding: binary
Content-ID: <ZvSvkz@pres.example.com>
Content-Type: application/pidf+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns: rp="urn:ietf:params:xml:ns:pidf:rpid"
             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             entity="pres:user3_public1@home3.net">

     <tuple id="h7833hjkk.dsajfjdsaf">
       <status>
         <basic>closed</basic>
       </status>
       <rp:class>sip</rp:class>
       <rp:privacy><rp:text/></rp:privacy>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user3_public1@home3.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="hu">Senki se merjen zavarni!</note>
       <timestamp>2003-08-27T11:48:59Z</timestamp>
     </tuple>

     <tuple id="sajdhdsahjh75vvcb774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>supervisor</rp:class>
       <rp:relationship><rp:supervisor/></rp:relationship>
       <contact priority="1.0">tel:+1-858-204-9141</contact>
       <note xml:lang="en">He's my supervisor</note>
       <timestamp>2003-08-27T11:48:59Z</timestamp>
     </tuple>

     <dm:person>
       <ci:homepage>http://example.com/~user3</ci:homepage>
       <ci:card>http://example.com/~user3/card.vcd</ci:card>
       <rp:class>presentity</rp:class>
       <rp:activities><rp:vacation/></rp:activities>
       <rp:place-type until="2003-09-10T17:30:00Z"><rp:ship/></rp:place-type>
     </dm:person>

   </presence>

--50UBfW7LSCVLtggUPe5z--
```

P-Charging-Vector: The RLS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

Content-Type: Set to the value of the Accept: header received in the SUBSCRIBE request.

The message body in the NOTIFY request that carries the presence information of the presentity is formed as indicated in draft-ietf-simple-event-list [22], draft-ietf-simple-presence-data-model [44], draft-ietf-simple-rpid [26], draft-ietf-simple-cipid [32] and draft-ietf-simple-prescaps-ext [25].

20. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.3.2-20**

The S-CSCF#1 forwards the NOTIFY request to the P-CSCF.

Table A.3.3.2-20: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the P-CSCF.

21. **NOTIFY request (P-CSCF to UE) - see example in table A.3.3.2-21**

The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.3.2-21: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1SIP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

22. **200 (OK) response (UE to P-CSCF) - see example in table A.3.3.2-22**

The UE acknowledges the NOTIFY request with a 200 (OK) to the P-CSCF.

Table A.3.3.2-22: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1SIP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

23. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.3.2-23**

The P-CSCF forwards the 200 (OK) response to the S-CSCF#1.

Table A.3.3.2-23: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home2.net;branch=z9hG4bK240f34.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024", orig-ioi=hom1.net,
      term-ioi=visited1.net
P-Access-Network-Info:
From:
To:
Call-ID:
CSeq:
Content-Length:
```

24. **200 (OK) response (S-CSCF to RLS) – see example in table A.3.3.2-24**

The S-CSCF#2 forwards the response to the RLS in the home network of the UE.

Table A.3.3.2-24: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home2.net;branch=z9hG4bK240f34.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024", orig-ioi=hom1.net;
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

A.3.4 RLS subscribing to presentities in different network

A.3.4.1 Successful subscription

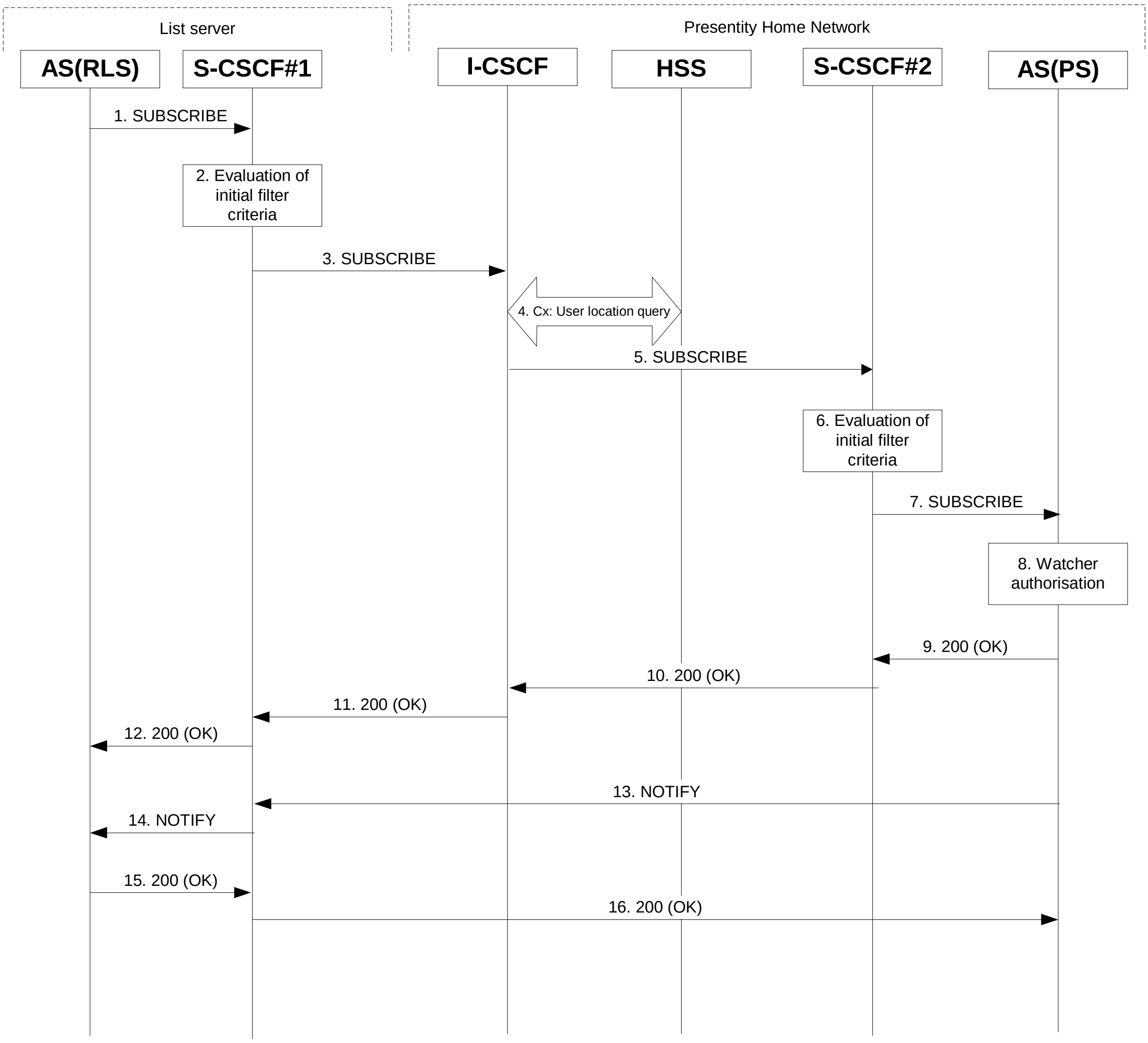

Figure A.3.4.1-1 RLS subscribing to presentities in different network

Figure A.3.4.1-1 shows the RLS subscribing to presence event notification about a presentity. The presentity is in a different IM CN subsystem. The details of the signalling flows are as follows:

1. **SUBSCRIBE request (RLS to S-CSCF) – see example in table A.3.4.1-1**

   The RLS resolves the watcher's resource address (the address is received according to subclause A.3.3) and subscribes to presence event notification at all the presentities that are represented by the resource list SIP URI. The home network of these presentities can be different or in the same network, as the RLS. In this example only a single subscription is shown where the home network of the presentity is another network. Subscriptions to other presentities follow a similar procedure. To initiate a subscription, the RLS generates a SUBSCRIBE request containing the "presence" event that it wishes to be notified of, together with an indication of the length of time this periodic subscription should last. The RLS sends the SUBSCRIBE request to the S-CSCF of "sip:user1_public1@home1.net" (S-CSCF#1). The address of S-CSCF#1 is either remembered from previous transactions (when "sip:user1_public1@home1.net" has subscribed for the resource list) or queried by the RLS using the Sh interface.

Table A.3.4.1-1 SUBSCRIBE request (RLS to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bKehuefdam
Max-Forwards: 70
Route: <sip:scscf1.home1.net;lr>
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user2_public1@home2.net>
Call-ID: q987a9a87g087abgf7qyg7ag
CSeq: 123 SUBSCRIBE
Event: presence
Expires: 7200
Accept: application/pidf+xml
Contact: <sip:rls.home1.net>
Content-Length: 0
```

Request-URI: Public user identity whose events the RLS subscribes to.

P-Charging-Vector: The RLS populates the icid parameter with a new globally unique value and populates the originating Inter Operator Identifier (IOI) parameter with the identifier of its own network of RLS.

P-Charging-Function-Addresses: The RLS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

To: Same as the Request-URI.

Event: This field is populated with the value "presence" to specify the use of the presence package.

Accept: This field is populated with the value "application/pidf+xml".

2. **Evaluation of initial filter criteria**

   S-CSCF#1 validates the service profile of this subscriber and evaluates the initial filter criteria. In this example, no AS is assumed to be involved.

3. **SUBSCRIBE request (S-CSCF to I-CSCF) – see example in table A.3.4.1-3**

   S-CSCF#1 performs an analysis of the destination address, and determines the network operator to whom the destination subscriber belongs. S-CSCF#1 forwards the request to the I-CSCF.

Table A.3.4.1-3 SUBSCRIBE request (S-CSCF to I-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
Max-Forwards: 69
Record-Route: <sip:orig@scscf1.home1.net;lr>
P-Asserted-Identity:
P-Charging-Vector:
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received.

4. **Cx: User Location Query procedure**

The I-CSCF sends a query to the HSS to find out the S-CSCF of the presentity. The HSS responds with the address of the current S-CSCF for the presentity.

For detailed message flows see 3GPP TS 29.228 [10].

Table A.3.4.1-4a provides the parameters in the SIP SUBSCRIBE request (flow 3), which are sent to the HSS.

Table A.3.4.1-4a: Cx: User registration status query procedure (I-CSCF to HSS)

Message source and destination	Cx: Information element name	Information source in SIP SUBSCRIBE	Description
I-CSCF to HSS	User Public Identity	Request-URI	This information element indicates the public user identity

Table A.3.4.1-4b provides the parameters sent from the HSS that need to be mapped to SIP SUBSCRIBE request (flow 5) and sent to the S-CSCF.

Table A.3.4.1-4b: Cx: User registration status query procedure (HSS to I-CSCF)

Message source and destination	Cx: Information element name	Mapping to SIP header in SIP SUBSCRIBE	Description
HSS to I-CSCF	S-CSCF name	Route header field	This information indicates the serving CSCF's name of that user

5. **SUBSCRIBE request (I-CSCF to S-CSCF) - see example in table A.3.4.1-5**

The I-CSCF forwards the SUBSCRIBE request to the S-CSCF#2 that will handle the termination.

Table A.3.4.1-5: SUBSCRIBE request (I-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bKj5hgrt2o, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
Max-Forwards: 68
P-Asserted-Identity:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net;
Route: <sip:scscf2.home2.net;lr>
Record-Route:
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

6. **Evaluation of initial filter criteria**

S-CSCF#2 validates the service profile of this subscriber and evaluates the initial filter criteria. For sip:user2_public1@home2.net the S-CSCF has Termination initial Filter Criteria with Service Point Trigger of Method = SUBSCRIBE AND Event = "presence" that informs the S-CSCF to route the SUBSCRIBE request to the AS ps.home2.net. The S-CSCF#2 has preconfigured information not to create a Record-Route entry for this request.

7. **SUBSCRIBE request (S-CSCF to PS) - see example in table A.3.4.1-7**

The S-CSCF#2 forwards the SUBSCRIBE request to the PS.

Table A.3.4.1-7 SUBSCRIBE request (S-CSCF to PS)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      icscf2_s.home2.net;branch=z9hG4bKj5hgrt2o, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
Max-Forwards: 67
P-Asserted-Identity:
P-Charging-Vector:
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:ps.home2.net;lr>, <sip:scscf2.home2.net;lr>
Record-Route: <sip:orig@scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the PS.

8. **Authorization of watcher**

The PS performs the necessary authorization checks on the originator to ensure it is allowed to watch the presentity. In this example all privacy conditions are met, so the PS sends a 200 (OK) response to the S-CSCF.

In the case where the privacy/authorization checks fails, then a necessary 2xx or 4xx response will be sent to the S-CSCF. The selection of the correct response code depends on the presentity's subscription authorization policy document.

9. **200 (OK) response (PS to S-CSCF) - see example in table A.3.4.1-9**

The PS sends the response to S-CSCF#2.

Table A.3.4.1-9: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      icscf2_s.home2.net;branch=z9hG4bKj5hgrt2o, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-
      ioi=home2.net:term-ioi=home2.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route:
From:
To: <sip:user2_public1@home2.net>;tag=151170
Call-ID:
CSeq:
Expires:
Contact: <sip:ps.home2.net;lr>
Content-Length: 0
```

P-Charging-Vector: The PS stores the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS stores the P-Charging-Function-Addresses header field and passes this header to the S-CSCF.

10. **200 (OK) response (S-CSCF to I-CSCF) - see example in table A.3.4.1-10**

S-CSCF#2 forwards the response to the I-CSCF.

Table A.3.4.1-10: 200 (OK) response (S-CSCF to I-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bKj5hgrt2o, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
P-Charging-Vector:
P-Charging-Function-Addresses:
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the I-CSCF.

11. **200 (OK) response (I-CSCF to S-CSCF) - see example in table A.3.4.1-11**

The I-CSCF forwards the response to S-CSCF#1.

Table A.3.4.1-11: 200 (OK) response (I-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bKehuefdam
P-Charging-Vector:
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

12. **200 (OK) response (S-CSCF to RLS) - see example in table A.3.4.1-12**

S-CSCF#1 forwards the response to the RLS.

Table A.3.4.1-12: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bKehuefdam
P-Charging-Vector:
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the RLS.

13. **NOTIFY request (PS to S-CSCF) - see example in table A.3.4.1-13**

As soon as the PS sends a 200 (OK) response to accept the subscription, it sends a NOTIFY request with the current state of the presentity's presence information that the watcher has subscribed and been authorized to. The NOTIFY request is sent to S-CSCF#1.

Table A.3.4.1-13: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:rls.home1.net SIP/2.0
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 70
Route: <sip:scscf1.home1.net;lr>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home2.net
From: <sip:user1_public1@home2.net>;tag=151170
To: <sip:rls.home1.net>;tag=31415
Call-ID: q987a9a87g087abgf7qyg7ag
CSeq: 42 NOTIFY
Subscription-State:active;expires=7200
Event: presence
Contact: <sip:ps.home2.net>
Content-Type: application/pidf+xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"

             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
        entity="pres:user2_public1@home2.net ">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>sip</rp:class>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">im:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>assistant</rp:class>
       <rp:relationship><rp:assistant/></rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person>
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>

       <rp:activities><rp:meeting/></rp:activities>
       <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>

     </dm:person>

   </presence>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

Content-Type: Set to the value of the Accept header received in the SUBSCRIBE request or "application/pidf+xml".

The message body in the NOTIFY request that carries the presentity's presence state is formed as indicated in RFC 3863 [21], draft-ietf-simple-presence-data-model [44], draft-ietf-simple-rpid [26], draft-ietf-simple-cipid [32] and draft-ietf-simple-prescaps-ext [25].

14. **NOTIFY request (S-CSCF to RLS) - see example in table A.3.4.1-14**

The S-CSCF#1 forwards the NOTIFY request to the RLS.

Table A.3.4.1-14: NOTIFY request (S-CSCF to RLS)

```
NOTIFY sip:rls.home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 69
P-Charging-Vector:
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the RLS.

15. **200 (OK) response (RLS to S-CSCF) – see example in table A.3.4.1-15**

The RLS generates a 200 (OK) response to the NOTIFY request.

Table A.3.4.1-15: 200 (OK) response (RLS to S-CSCF)

```
SIP/2.0 200 OK
Via:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-
      ioi=home1.net:term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

P-Charging-Vector: The RLS stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

16. **200 (OK) response (S-CSCF to PS) – see example in table A.3.4.1-16**

The S-CSCF#1 forwards the 200 (OK) response to the PS.

Table A.3.4.1-16: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
P-Charging-Vector:
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

A.3.5 Network based watcher subscribing on behalf of IMS watcher to IMS presentities

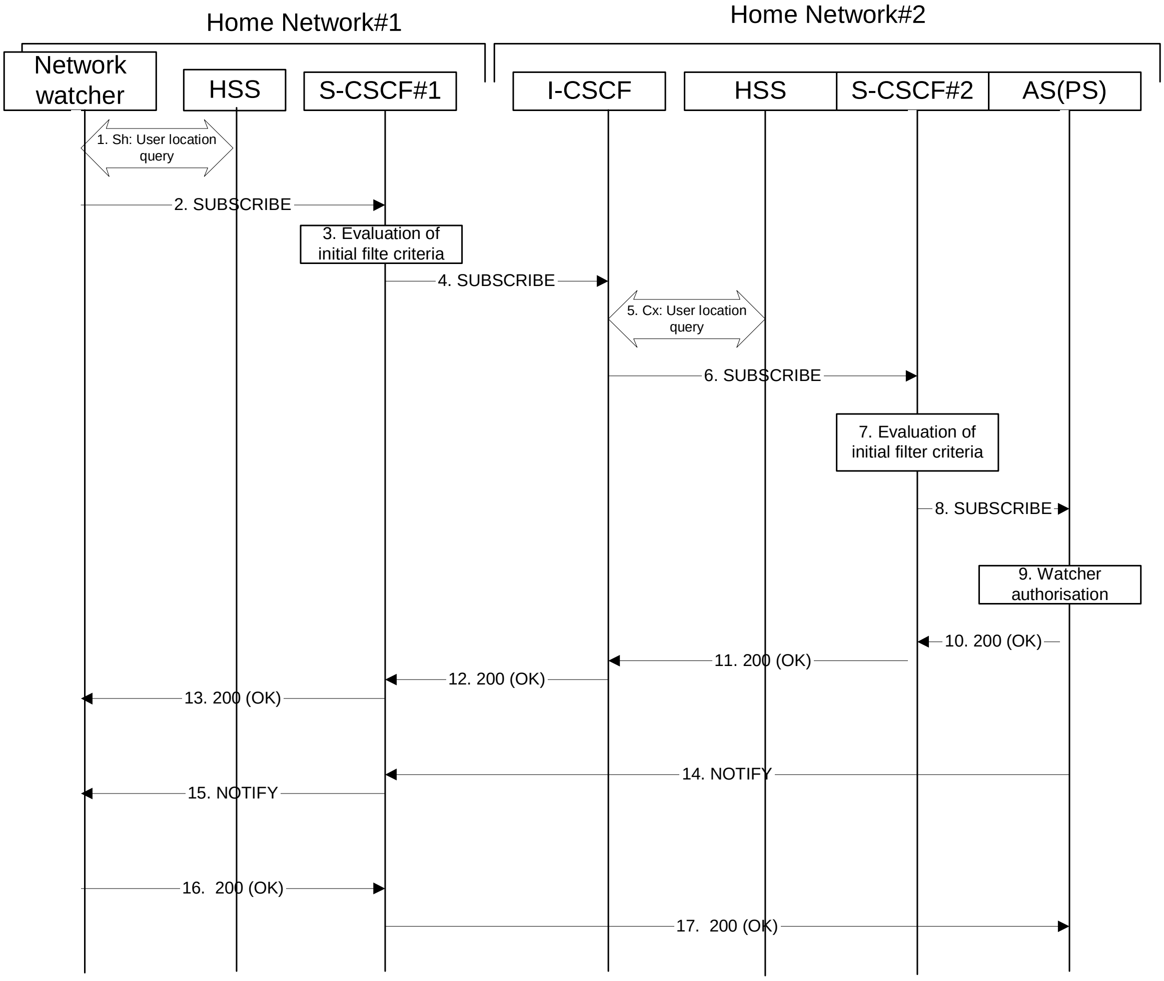

Figure A.3.5-1: Network based watcher subscribing on behalf of IMS watcher for presence information of IMS presentities

Figure A.3.5-1 shows a trusted network based watcher subscribing on behalf of an IMS watcher to presence event notification about an IMS based presentity. The presentity is in a different IM CN subsystem than the network based watcher and the signalling flow assumes that the IMS watcher on whose behalf the network based watcher subscribes is registered to the IMS network. The details of the signalling flows are as follows:

1. **Sh: User Location Query procedure**

   The network based watcher sends a query to the HSS to find out the S-CSCF of the user on whose behalf the subscription is initiated. The HSS responds with the address of the current S-CSCF for the originating subscriber.

2. **SUBSCRIBE request (Network based watcher to S-CSCF) - see example in table A.3.5-2**

   The SUBSCRIBE request is constructed and forwarded to S-CSCF. The S-CSCF is inserted into the Route header of the SUBSCRIBE request.

Table A.3.5-2: SUBSCRIBE request (network watcher to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP watcher.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
Privacy: none
Route: <sip:scscf1.home1.net;lr;orig>
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user2_public1@home2.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 61 SUBSCRIBE
Event:PRESENCE
Expires: 7200
Accept: application/pidf+xml;q=0.3, application/pidf-diff+xml;q=1
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request-URI: Public user identity of the user to whose events the subscriber subscribes to.

P-Asserted-Identity: The network based watcher inserts the public user identity of the watcher on whose behalf the subscription is made into the P-Asserted-Identity header field.

Route: The Route header is populated with the address of the S-CSCF obtained from the response to the user location query performed by the network based watcher on the Sh interface.

Event: This field is populated with the value "presence" to specify the use of the presence package.

Contact: The contact information of the network based watcher.

3. **Evaluation of initial filter criteria**

   S-CSCF#1 validates the service profile of the subscriber identified in the P-Asserted-Identity header field and evaluates the initial filter criteria. In this example, no AS is assumed to be involved.

4. **SUBSCRIBE request (S-CSCF to I-CSCF) - see example in table A.3.5-4**

   S-CSCF#1 performs an analysis of the destination address, and determines the network operator to whom the destination subscriber belongs. Since the originating operator does not desire to keep their internal configuration hidden, S-CSCF#1 forwards the SUBSCRIBE request directly to the I-CSCF in the destination network.

Table A.3.5-4: SUBSCRIBE (S-CSCF to I-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1,
Max-Forwards: 68
P-Asserted-Identity: <sip:user1_public1@home1.net>
Privacy:
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

5. **Cx: User Location Query procedure**

The I-CSCF sends a query to the HSS to find out the S-CSCF of the called user. The HSS responds with the address of the current S-CSCF for the terminating subscriber.

For detailed message flows see 3GPP TS 29.228 [10].

Table A.3.5-5a provides the parameters in the SIP SUBSCRIBE request (flow 4), which are sent to the HSS.

Table A.3.5-5a: Cx: User registration status query procedure (I-CSCF to HSS)

Message source and destination	Cx: Information element name	Information source in SIP SUBSCRIBE	Description
I-CSCF to HSS	User Public Identity	Request-URI	This information element indicates the public user identity

Table A.3.5-5b provides the parameters sent from the HSS that need to be mapped to the SIP SUBSCRIBE request (flow 6) and sent to the S-CSCF.

Table A.3.5-5b: Cx: User registration status query procedure (HSS to I-CSCF)

Message source and destination	Cx: Information element name	Mapping to SIP header in SIP SUBSCRIBE	Description
HSS to I-CSCF	S-CSCF name	Route header field	This information indicates the serving CSCF's name of that user

6. **SUBSCRIBE request (I-CSCF to S-CSCF) - see example in table A.3.5-6**

The I-CSCF forwards the SUBSCRIBE request to the S-CSCF (S-CSCF#2) that will handle the termination.

Table A.3.5-6: SUBSCRIBE request (I-CSCF to S-CSCF)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 67
P-Asserted-Identity:
Privacy:
Route: <sip:scscf2.home2.net;lr>
Record-Route:
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

NOTE: The I-CSCF does not add itself to the Record-Route header, as it has no need to remain in the signalling path for the subsequent requests.

7. **Evaluation of initial filter criteria**

   S-CSCF#2 validates the service profile of this subscriber and evaluates the initial filter criteria. For sip:user2_public1@home2.net S-CSCF#2 has termination initial filter criteria with Service Point Trigger of Method = SUBSCRIBE and Event = "presence" that informs the S-CSCF to route the SUBSCRIBE request to the AS ps.home2.net. The S-CSCF#2 has preconfigured information not to create a Record-Route header for this request.

8. **SUBSCRIBE request (S-CSCF to PS) - see example in table A.3.5-8**

   The S-CSCF forwards the SUBSCRIBE request to the PS.

Table A.3.5-8: SUBSCRIBE request (S-CSCF to PS)

```
SUBSCRIBE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
      icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 66
P-Asserted-Identity:
Privacy:
Route: <sip:ps.home2.net;lr>, <sip:scscf2.home2.net;lr>
Record-Route: <sip: scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

9. **Authorization of watcher**

   The PS performs the necessary authorization checks on the watcher whose behalf the subscription is being made to ensure it is allowed to watch the presentity. In this example all privacy conditions are met, so the PS sends a 200 (OK) response to the S-CSCF.

   In the case where the privacy/authorization checks fails, then a necessary 2xx or 4xx response will be sent to the S-CSCF. The selection of the correct response code depends on the presentity's authorization policy document.

10. **200 (OK) response (PS to S-CSCF) - see example in table A.3.5-10**

The PS sends the response to S-CSCF#2.

Table A.3.5-10: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1
Record-Route:
From:
To: <sip:user2_public1@home2.net>;tag=151170
Call-ID:
CSeq:
Expires:
Contact: <sip:ps.home2.net>
Content-Length:
```

11. **200 (OK) response (S-CSCF to I-CSCF) - see example in table A.3.5-11**

S-CSCF#2 forwards the response to I-CSCF#2.

Table A.3.5-11: 200 (OK) response (S-CSCF to I-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

12. **200 (OK) response (I-CSCF to S-CSCF) - see example in table A.3.5-12**

I-CSCF#2 forwards the response to S-CSCF#1.

Table A.3.5-12: 200 (OK) response (I-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      network.home1.net;branch=z9hG4bK240f34.1
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

13. **200 (OK) response (S-CSCF to network watcher) - see example in table A.3.5-13**

S-CSCF#1 forwards the response to request originator.

Table A.3.5-13: 200 (OK) response (S-CSCF to network watcher)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP network.home1.net;branch=z9hG4bK240f34.1
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

14. **NOTIFY request (PS to S-CSCF) - see example in table A.3.5-14**

As soon as the PS sends a 200 (OK) response to accept the subscription, it sends a NOTIFY request with the current state of the presentity's presence information that the watcher has subscribed and been authorized to. The NOTIFY request is sent to S-CSCF#1. Based on the Accept header field of the SUBSCRIBE request, the PS decides to use partial notifications to provide further changes of presence information. The first notification always contains the full state. The 'application/pidf-diff +xml' content type is used.

Table A.3.5-14: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip: network.home1.net;branch=z9hG4bK240f34.1 SIP/2.0
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 70
Route: <sip:scscf1.home1.net;lr>
From: <sip:user2_public1@home2.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 42 NOTIFY
Subscription-State: active; expires=7200
Event: presence
Contact: <sip:ps.home2.net>
Content-Type: application/pidf-diff +xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   < diff:pidf-full xmlns="urn:ietf:params:xml:ns:pidf"
             xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"
             xmlns:diff="urn:ietf:params:xml:ns:pidf:pidf-diff"
      xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
             entity="pres:user2_public1@home2.net"version="1">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
         <rp:privacy><text/></rp:privacy>
         <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       </status>
       <rp:class>sip</rp:class>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>assistant</rp:class>
       <rp:relationship><rp:assistant/></rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person>
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>
       <rp:activities><rp:meeting/></rp:activities>
       <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>
     </dm:person>

   </diff:pidf-full>
```

From: The tag of this field matches that of the To field in the received 200 (OK) response for the SUBSCRIBE request.

Content-Type: Set to the preferred value of the Accept header received in the SUBSCRIBE request.

The message body in the NOTIFY request that carries the presence information of the presentity is formed as indicated in RFC 3863 [21], draft-ietf-simple-rpid [26], draft-ietf-simple-cipid [32], draft-ietf-simple-prescaps-ext[25], draft-ietf-simple-presence-data-model [44] and draft-ietf-simple-partial-notify [24].

15. **NOTIFY request (S-CSCF to network watcher) - see example in table A.3.5-15**

The S-CSCF#1 forwards the NOTIFY request to the network watcher

Table A.3.5-15: NOTIFY request (S-CSCF to network watcher)

```
NOTIFY sip: network.home1.net;branch=z9hG4bK240f34.1SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 69
Privacy:
Record-Route: <sip:scscf1.home1.net;lr>

From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

16. **200 (OK) response (network watcher to S-CSCF) – see example in table A.3.5-16**

The network watcher forwards the 200 (OK) response to S-CSCF#1.

Table A.3.5-16: 200 (OK) response (network watcher to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info:
From:
To:
Call-ID:
CSeq:
Content-Length:
```

17. **200 (OK) response (S-CSCF to PS) – see example in table A.3.5-17**

S-CSCF#2 forwards the 200 (OK) response to the PS.

Table A.3.5-17: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
From:
To:
Call-ID:
CSeq:
Content-Length:
```

A.3.6 Watcher subscribing to state changes in XML document, UE in visited network

A.3.6.1 Watcher subscribing to changes made via XCAP in his resource list, UE in visited network - Successful subscription

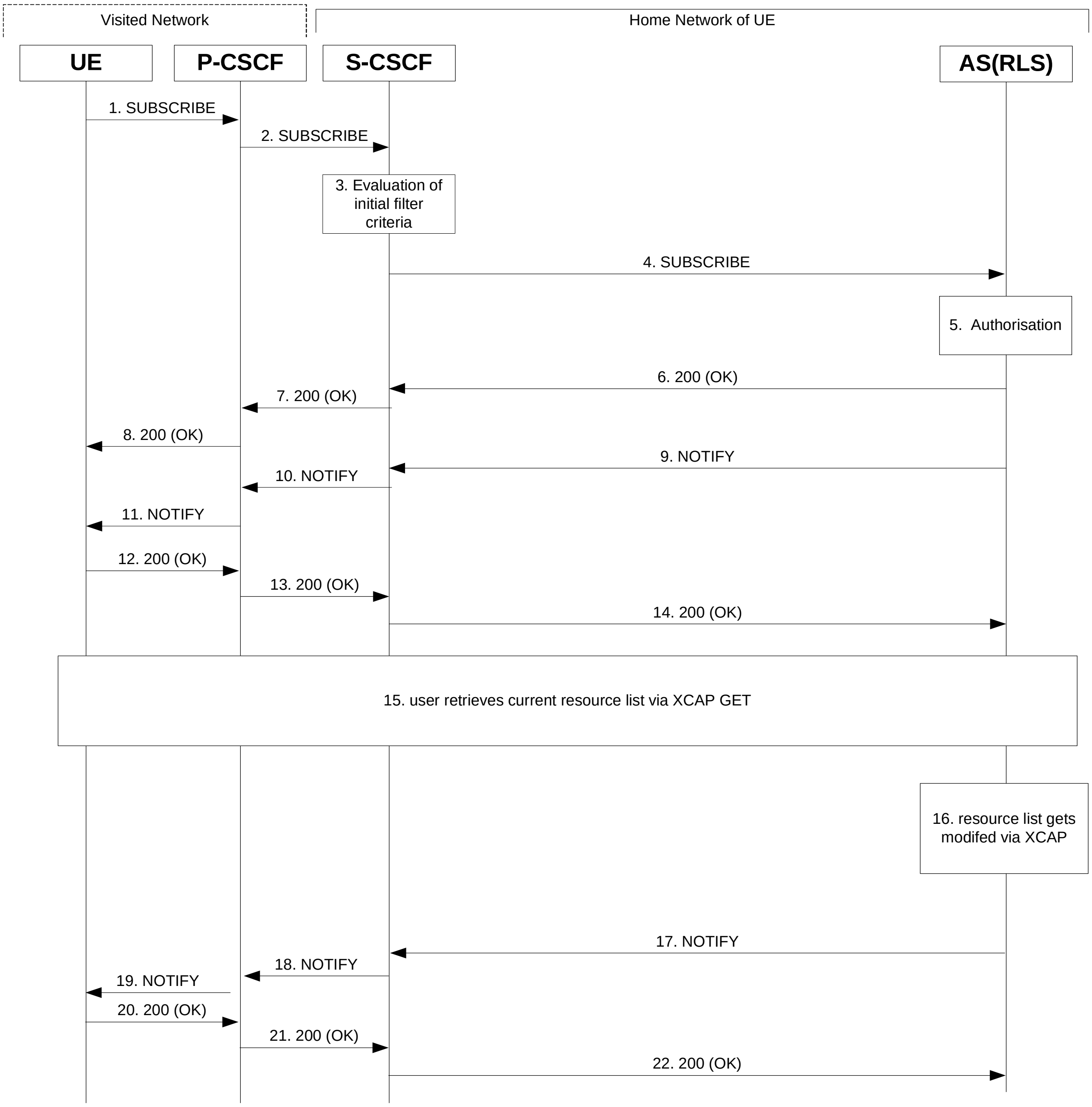

Figure A.3.6.1-1: Watcher subscribing to changes made via XCAP in his resource list

Figure A.3.6.1-1 shows a watcher subscribing to notifications of state changes made via XCAP in his resource list. The details of the flows as follows:

1. **SUBSCRIBE request (UE to P-CSCF) – see example in table A.3.6.1-1**

   A watcher agent in a UE wishes to get notification when his resource list gets modified via XCAP. In order to initiate a subscription to XCAP document changes in RLS, the UE generates a SUBSCRIBE request indicating support for "ua-profile", together with an indication of the length of time this periodic subscription should last.

Table A.3.6.1-1: SUBSCRIBE request (UE to P-CSCF)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user1_public1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 123 SUBSCRIBE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: ua-profile;profile-type=application;app-id=resource-lists;document="users/user1"
Expires: 7200
Accept: application/xcap-diff+xml
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request-URI: The users own SIP URI to get notifications of changes on all lists owned by the user.

Event: This field is populated with the value "ua-profile" to specify the use of the ua-profile package to get notified of changes to XCAP documents. The "app-id" in the field identifies the XCAP application usage. The "document" further details the document that is being subscribed.

Accept: This field is populated with the value "application/xcap-diff+xml" indicating that the UE supports the MIME type.

To: Same as the Request-URI.

2. **SUBSCRIBE request (P-CSCF to S-CSCF) - see example in table A.3.6.1-2**

   The P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The SUBSCRIBE request is forwarded to S-CSCF#1. A Route header is inserted into SUBSCRIBE request. The information for the Route header is taken from the service route determined during registration.

Table A.3.6.1-2: SUBSCRIBE request (P-CSCF to S-CSCF)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Access-Network-Info:
Route: <sip:orig@scscf1.home1.net;lr>
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
Privacy:
Record-Route: <sip:pcscf1.visited1.net;lr>
Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

3. **Evaluation of initial filter criteria**

The S-CSCF validates the service profile of this subscriber and evaluates the initial filter criteria. For sip:user1_public1@home1.net the S-CSCF has originating initial Filter Criteria with Service Point Trigger of Method = SUBSCRIBE AND Event = "ua-profile" that informs the S-CSCF to route the SUBSCRIBE request to the AS sip:rls.home1.net.

4. **SUBSCRIBE request (S-CSCF to RLS) - see example in table A.3.6.1-4**

The S-CSCF forwards the SUBSCRIBE request to the RLS.

Table A.3.6.1-4 SUBSCRIBE request (S-CSCF to RLS)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 68
P-Access-Network-Info:
P-Asserted-Identity: <sip:user1_public1@home1.net>, <tel:+1-212-555-1111>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
Route: <sip:rls.home1.net;lr>, <sip:orig@scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Supported:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the RLS.

5. **Authorization**

The RLS performs the necessary authorization checks on the originator to ensure that he/she is authorized to subscribe to XML document changes. In this example this condition has been met, so the RLS sends a 200 (OK) response to the S-CSCF.

6. **200 (OK) response (RLS to S-CSCF) - see example in table A.3.6.1-6**

The RLS sends the response to the S-CSCF.

Table A.3.6.1-6: 200 (OK) response (RLS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net;
      term-ioi=home1.net
Record-Route:
From:
To: <sip:user1_public1@home1.net>;tag=151170
Call-ID:
CSeq:
Expires:
Contact:
Content-Length: 0
```

7. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.3.6.1-7**

The S-CSCF forwards the response to the P-CSCF.

Table A.3.6.1-7: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

8. **200 (OK) response (P-CSCF to UE) - see example in table A.3.6.1-8**

The P-CSCF forwards the response to the watcher agent in the UE.

Table A.3.6.1-8: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

9. **NOTIFY request (RLS to S-CSCF) – see example in table A.3.6.1-9**

The RLS generates a NOTIFY request including the xcap-diff document as a result of the SUBSCRIBE request. As this is the initial NOTIFY it contains only the new-etag, previous-etag and document-selector elements.

Table A.3.6.1-9 NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_@home1.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 89 NOTIFY
Subscription-State: active;expires=7200
Event: ua-profile
Contact: <sip:rls.home1.net>
Content-Type: application/xcap-diff+xml;charset="UTF-8"
Content-Length:

<?xml version="1.0" encoding="UTF-8"?>

<xcap-diff xmlns="urn:ietf:params:xml:ns:xcap-diff"

         xcap-root="http://xcap.home1.net/services">

    <document doc-selector="resource-lists/users/user1/friends"
           new-etag="7hahsd"/>
    </document>
    <document doc-selector="resource-lists/users/user1/coworkers"
              new-etag="ffds66a">
    </document>

</xcap-diff>
```

The content of the document element contains a new-etag and a previous etag element with identical value and no list of instructions. This way it is indicated that this is the reference XML diff document. This documents has only the information about the etags and the document URI"s covered by that subscription.

10. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.6.1-10**

    The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.3.6.1-10: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Length:

(…)
```

11. **NOTIFY request (P-CSCF to UE) - see example in table A.3.6.1-11**

    The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.6.1-11: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Length:

(…)
```

12. **200 (OK) response (UE to P-CSCF) - see example in table A.3.6.1-12**

    The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.3.6.1-12: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

13. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.6.1-13**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.3.6.1-13: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

14. **200 (OK) response (S-CSCF to RLS) - see example in table A.3.6.1-14**

    The S-CSCF#2 forwards the response to the RLS in the home network of the UE.

Table A.3.6.1-14: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net;
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

15. **User retrieves current resource list via XCAP get**

    As user1 does not have a local copy of the resource list identified by the etag he retrieves the corresponding list via XCAP get.

16. **Resource List gets modified via XCAP**

    The resource list of user1 gets modified via XCAP procedures.

17. **NOTIFY request (RLS to S-CSCF) - see example in table A.3.6.1-17**

    In this example it is assumed that the RLS has received a XCAP request to delete user2_public@home1.net from the resource list of user1.

Table A.3.6.1-17 NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 90 NOTIFY
Subscription-State: active;expires=5000
Event: ua-profile
Contact: <sip:rls.home1.net>
Content-Type: application/xcap-diff+xml;charset="UTF-8"
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>

<xcap-diff xmlns="urn:ietf:params:xml:ns:xcap-diff"
        xcap-root="http://xcap.home1.net/services">

    <document doc-selector="resource-lists/users/user1/coworkers"
              new-etag="aaaab" previous-etag="ffds66a">
      <change-log>
        <put-eventnode-selector="resource-lists/list[@name=&quot;coworkers&quot;]" content-
      type="application/el+xml">
          <![CDATA[<entry uri="sip:new-worker@example.com"/>]]>
        </put-event>
      </change-log>
    </document>
</xcap-diff>
```

Content-Type: Set to application/xcap-diff+xml.

The message body in the NOTIFY request contains information of the new-etag of the changed document, the change method and the element that was changed in accordance with draft-ietf-simple-xcap-diff [39].

18. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.3.6.1-18**

    The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.3.6.1-18: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content Type:
Content-Length:

(…)
```

19. **NOTIFY request (P-CSCF to UE) – see example in table A.3.6.1-19**

    The P-CSCF forwards the NOTIFY request to the watcher in the UE.

Table A.3.6.1-19: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Content-Type:
Content-Length:

(…)
```

20. **200 (OK) response (UE to P-CSCF) – see example in table A.3.6.1-20**

    The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.3.6.1-20: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

21. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.3.6.1-21**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.3.6.1-21: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

22. **200 (OK) response (S-CSCF to RLS) – see example in table A.3.6.1-22**

    The S-CSCF#2 forwards the response to the RLS in the home network of the UE.

Table A.3.6.1-22: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home1.net;
      term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

A.4 Signalling flows demonstrating how presentities update presence information

A.4.1 Introduction

This subclause covers the signalling flows that show how presentities update presence information in the PS.

A.4.2 Initial publication or modification of presence information by UE

A.4.2.1 Successful publication

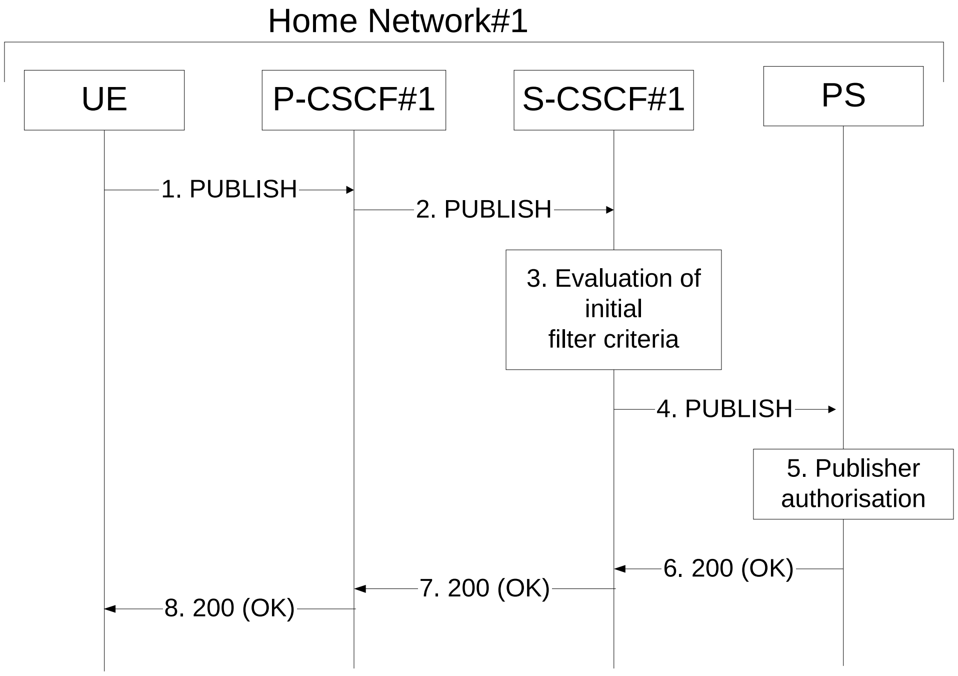

Figure A.4.2.1-1: UE publishing presence information

The UE may publish the partial presence information or the full presence information about a presentity to the PS.

In this example, it is assumed that the UE publishes the full presence information.

Figure A.4.2.1-1 shows a UE publishing or modifying already existing presence information about a presentity. The details of the signalling flows as follows:

1. **PUBLISH request (UE to P-CSCF) - see example in table A.4.2.1-1**

   A PUA in a UE wishes to publish presence information. To initiate the publication, the UE generates a PUBLISH request according to RFC 3903 [23] containing the presence information that it wishes to publish.

   The message body in the PUBLISH request that carries the PUA presence update state is formed as indicated in draft-ietf-simple-presence-data-model [44], RFC 3863 [21], draft-ietf-simple-rpid [26], draft-ietf-simple-cipid[32], and draft-ietf-simple-prescaps-ext [25].

Table A.4.2.1-1: PUBLISH request (UE to P-CSCF)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user1_public1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 61 PUBLISH
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: presence
Expires: 7200
Content-Type: application/pidf+xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
"
             xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"

             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
             entity="pres:user2_public1@home2.net">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>


       </status>
       <rp:class>sip</rp:class>
       <rp:privacy><text/></es:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</es:status-icon>
       <pcp:servcaps>
        <pcp:video>false</pcp:video>
        <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8">sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <tuple id="jklhgf9788934774.78">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>assistant</rp:class>
       <rp:relationship><rp:assistant/></rp:relationship>
       <contact priority="1.0">tel:+1-212-555-2222</contact>
       <note xml:lang="en">She's my secretary</note>
       <timestamp>2003-08-27T11:49:29Z</timestamp>
     </tuple>

     <dm:person>
```

```
        <rp:class>presentity</rp:class>
        <ci:homepage>http://example.com/~user2</ci:homepage>
        <ci:card>http://example.com/~user2/card.vcd</ci:card>
          <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>
      </dm:person>

    </presence>
```

Request-URI: Public user identity whose presence information the PUA intends to publish.

Event: This field is populated with the value "presence" to specify the use of the presence package.

To: Same as the Request-URI.

Content-Type: Set to the value "application/pidf+xml".

2. **PUBLISH request (P-CSCF to S-CSCF) - see example in table A.4.2.1-2**

   P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The PUBLISH request is forwarded to the S-CSCF. A Route header is inserted into PUBLISH request. The information for the Route header is taken from the service route determined during registration.

Table A.4.2.1-2: PUBLISH request (P-CSCF to S-CSCF)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Access-Network-Info:
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
Route: <sip:orig@scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Content-Type:
Content-Length:

(…)
```

3. **Evaluation of initial filter criteria**

   S-CSCF validates the service profile of the publisher and evaluates the initial filter criteria. For user1_public1@home1.net S-CSCF#1 has originating initial Filter Criteria with Service Point Trigger of Method = PUBLISH AND Event = "presence" AND Request-URI = "sip:user1_public1@home1.net" that informs the S-CSCF to route the PUBLISH request to the PS ps.home1.net.

4. **PUBLISH (S-CSCF to PS) - see example in table A.4.2.1-4**

   The S-CSCF#1 forwards the PUBLISH request to the PS.

Table A.4.2.1-4: PUBLISH request (S-CSCF to PS)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Access-Network-Info:
Max-Forwards: 68
P-Asserted-Identity:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Route: <sip:ps.home1.net;lr>, <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the PS.

5. **Authorization of publisher**

   The PS performs the necessary authorization checks on the originator to ensure it is allowed to publish the presentity's presence information. In this example all privacy conditions are met, so the PS sends a 200 (OK) response to the S-CSCF.

6. **200 (OK) response (PS to S-CSCF) - see example in table A.4.2.1-6**

   The PS sends the response to S-CSCF.

Table A.4.2.1-6: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-
      ioi=home1.net:term-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
From:
To: <sip:user1_public1@home1.net>;tag=151170
Call-ID:
CSeq:
Expires: 7200
SIP-ETag: 123xy
Content-Length: 0
```

P-Charging-Vector: The PS stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS stores the P-Charging-Function-Addresses header field and passes this header to the S-CSCF.

SIP-ETag: This field is populated with a locally unique entity-tag to associate further publications of this event state.

7. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.4.2.1-7**

S-CSCF forwards the response to P-CSCF.

Table A.4.2.1-7: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
P-Charging-Function-Addresses:
From:
To:
Call-ID:
CSeq:
Expires:
SIP-ETag:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter and removes the orig-ioi and the term-ioi parameters.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

8. **200 (OK) response (P-CSCF to UE) - see example in table A.4.2.1-6**

P-CSCF forwards the response to the PUA in the UE.

Table A.4.2.1-8: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
From:
To:
Call-ID:
CSeq:
Expires:
SIP-ETag:
Content-Length:
```

A.4.3 Refreshing of presence information by UE

A.4.3.1 Successful refresh

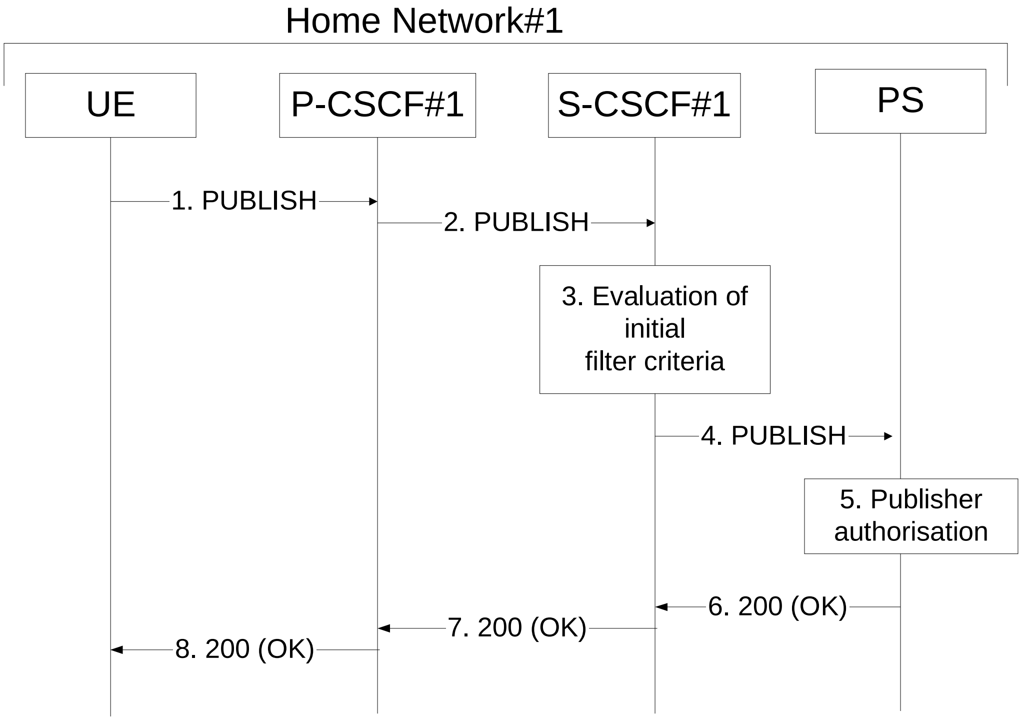

Figure A.4.3.1-1: UE updating presence information

Figure A.4.3.1-1 shows an UE refreshing the presence information about a presentity. The details of the signalling flows are as follows:

1. **PUBLISH request (UE to P-CSCF) – see example in table A.4.3.1-1**

   A PUA in a UE wishes to refresh already existing presence information. To initiate the publication, the UE generates a PUBLISH request according to RFC 3903 [23].

Table A.4.3.1-1: PUBLISH request (UE to P-CSCF)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user1_public1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a111
CSeq: 61 PUBLISH
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi-s=87654321; port-
      c=8642; port-s=7531
Event: presence
SIP-If-Match: 123xy
Expires: 7200
Content-Length: 0
```

Request-URI: Public user identity whose presence information the PUA intends to publish.

Event: This field is populated with the value "presence" to specify the use of the presence package.

To: Same as the Request-URI.

SIP-If-Match: This field is populated with the entity-tag earlier provided by the PS in the SIP-ETag header field of the previous 200(OK) response and is used as a versioning precondition to the PUBLISH refresh.

2. **PUBLISH request (P-CSCF to S-CSCF) - see example in table A.4.3.1-2**

   P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The PUBLISH request is forwarded to the S-CSCF. A Route header is inserted into PUBLISH request. The information for the Route header is taken from the service route determined during registration.

Table A.4.3.1-2: PUBLISH request (P-CSCF to S-CSCF)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Access-Network-Info:
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
Route: <sip:orig@scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
SIP-If-Match:
Expires:
Content-Length:
```

P-Charging-Vector: The P-CSCF populates the icid parameter with a globally unique value.

3. **Evaluation of initial filter criteria**

   S-CSCF#1 validates the service profile of this publisher and evaluates the initial filter criteria. For user1_public1@home1.net the S-CSCF has originating initial Filter Criteria with Service Point Trigger of Method = PUBLISH AND Event = "presence" AND Request-URI = "sip:user1_public1@home1.net" that informs the S-CSCF to route the PUBLISH request to the PS ps.home1.net.

4. **PUBLISH (S-CSCF to PS) – see example in table A.4.3.1-4**

   The S-CSCF forwards the PUBLISH request to the PS.

Table A.4.3.1-4: PUBLISH (S-CSCF to PS)

```
PUBLISH sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Access-Network-Info:
Max-Forwards: 68
P-Asserted-Identity:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Route: <sip:ps.home1.net;lr>, <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
SIP-If-Match:
Expires:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the PS.

5. **Authorization of publisher**

The PS performs the necessary authorization checks on the originator to ensure it is allowed to publish the presentity's presence information. In this example all privacy conditions are met, so the PS sends a 200 (OK) response to the S-CSCF.

6. **200 (OK) response (PS to S-CSCF) - see example in table A.4.3.1-6**

The PS sends the response to S-CSCF.

Table A.4.3.1-6: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP
      pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023555517"; orig-
      ioi=home1.net;term-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
From:
To: <sip:user1_public1@home1.net>;tag=151170
Call-ID:
CSeq:
Expires: 7200
SIP-ETag: 345abc
Content-Length: 0
```

P-Charging-Vector: The PS stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS stores the P-Charging-Function-Addresses header field and passes this header to the S-CSCF.

SIP-ETag: This field is populated with a new locally unique entity-tag.

7. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.4.3.1-7**

S-CSCF#1 forwards the response to P-CSCF.

Table A.4.3.1-7: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.home1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023555517"
P-Charging-Function-Addresses:
From:
To:
Call-ID:
CSeq:
Expires:
SIP-ETag:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter and removes the orig-ioi and the term-ioi parameters.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

8. **200 (OK) response (P-CSCF to UE) - see example in table A.4.3.1-6**

   P-CSCF#1 forwards the response to the PUA in the UE.

Table A.4.3.1-8: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
From:
To:
Call-ID:
CSeq:
Expires:
SIP-ETag:
Content-Length:
```

A.5 PS notifying watcher of updates to presence information

A.5.1 Introduction

This subclause covers the signalling flows that show how the PS notifies watchers of updates to presence information.

A.5.2 Watcher and presentity in the different networks, UE in the home network

A.5.2.1 Successful notification

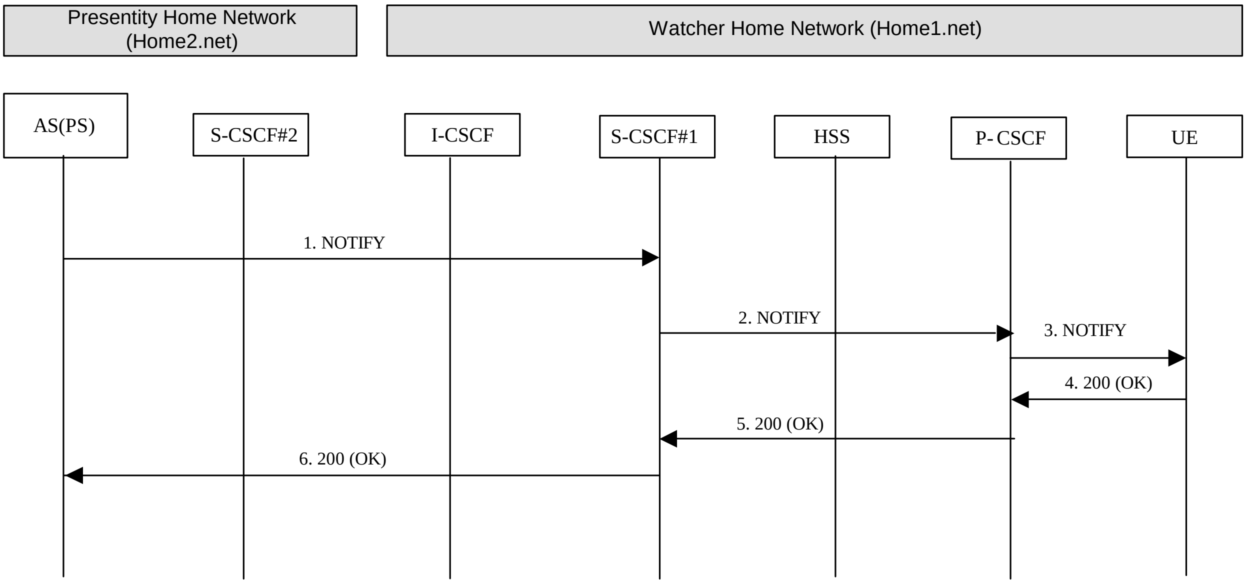

Figure A.5.2.1-1: Notification to watcher in the visited network

Figure A.5.2.1-1 shows how a watcher is notified of updates to a presentity's presence information. The signalling flow is applicable to the case where the watcher and presentity are in the same or in different IM CN subsystems.

1. **NOTIFY request (PS to S-CSCF) – see example in table A.5.2.1-1**

   The PS determines which authorized watchers are entitled to receive the updates of the presence information for this presentity. For each appropriate watcher, the PS sends a NOTIFY request that contains the updated state of presence information. The NOTIFY request may either contain the complete set of presence information, or only the information that has changed since the last notification. In this example, the watcher indicated preference for partial notification in the SUBSCRIBE request, so the NOTIFY request is formulated according to draft-ietf-simple-partial-notify [24] and draft-ietf-simple-partial-pidf-format [38] by including only the information that has changed since the last notification. (Note that the first NOTIFY request has contained the full state of the presence information.)

Table A.5.2.1-1: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=523551024"; orig-ioi=home2.net
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net;lr>
From: <sip:user2_public1@home2.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 43 NOTIFY
Subscription-State: active;expires=5000
Event: presence
Contact: <sip:ps.home2.net>
Content-Type: application/pidf-diff+xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   <pidf-diff xmlns="urn:ietf:params:xml:ns:pidf-diff"version="1"
          entity="pres:user2_public1@home2.net">

    <add parent="presence" sel="*[2]">
      <![CDATA[

      <tuple id="xfjsk">
       <status>
         <basic>open</basic>
       </status>
       <rp:class>voice</rp:class>
       <rp:status-icon>http://example.com/~user2/iconABC.gif</rp:status-icon>
       <contact priority="0.2">tel:40302020@home2.net</contact>
       <note xml:lang="en">This is a new tuple inserted as the 2nd tuple.</note>
       <timestamp>2004-11-01T11:49:29Z</timestamp>
      </tuple>
        ]] >
    </add>
    <replace sel="presence/tuple[@id="a8098a.672364762364"]/status/basic/text()">closed
    </replace>

    <remove sel="presence/dm:person[@id="89898"]/rp:privacy"/>

    <remove sel="presence/dm:person[@id="89898"]/rp:activities"/>

    <replace sel="presence/tuple[@id="a8098a.672364762364"]/rp:status-icon/text()">
    http://example.com/~user2/iconXYZ.gif</replace>

   </pidf-diff>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

2. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.5.2.1-2**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.5.2.1-2: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=523551024"
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:pcscf1.home1.net;lr>
Record-Route: <sip:scscf2.home2.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter.

P-Charging-Function-Addresses: The S-CSCF populates the P-Charging-Function-Addresses header field to be passed to the P-CSCF.

3. **NOTIFY request (P-CSCF to UE) - see example in table A.5.2.1-3**

   The P-CSCF forwards the NOTIFY request to the UE.

Table A.5.2.1-3: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.home1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

4. **200 (OK) response (UE to P-CSCF) - see example in table A.5.2.1-4**

   The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.5.2.1-4: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.home1.net;branch=240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

5. **200 (OK) response (P-CSCF to S-CSCF) - see example in table A.5.2.1-5**

   The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.5.2.1-5: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home2.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=523551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

6. **200 (OK) response (S-CSCF to PS) - see example in table A.5.2.1-6**

The S-CSCF forwards the 200 (OK) response to the PS.

Table A.5.2.1-6: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK240f34.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=523551024"; orig-
      ioi=home1.net:term-ioi=home2.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

A.5.3 Notification to resource list in a different network and notification to watcher in the visited network

A.5.3.1 Successful notification

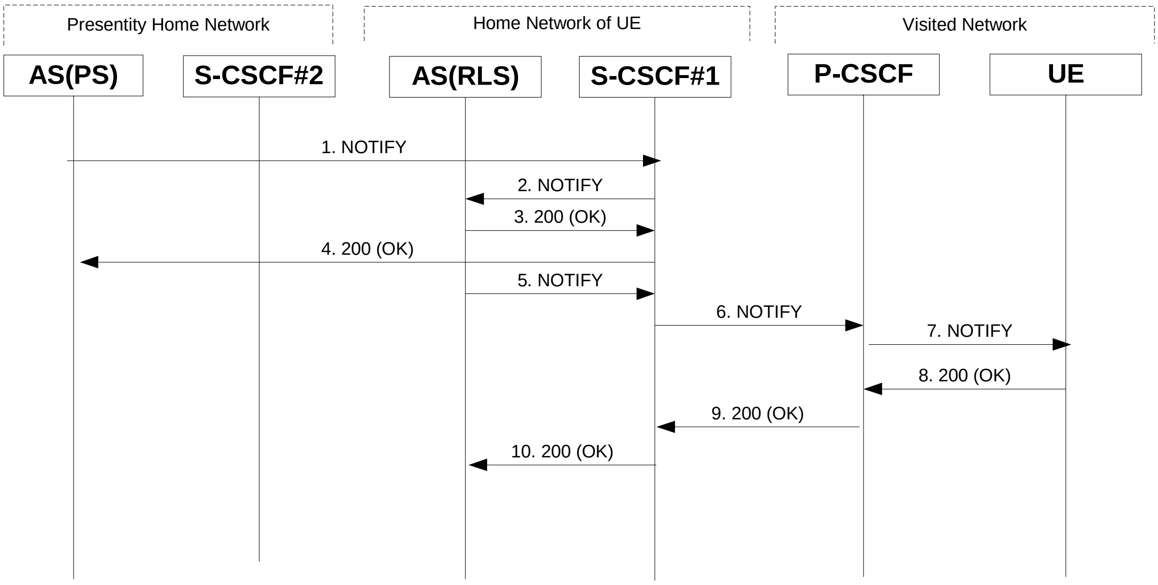

Figure A.5.3.1-1: Notification to resource list in a different network and notification to watcher in the visited network

Figure A.5.3.1-1 shows the PS providing presence event notification about a presentity to a RLS in a different network. This notification triggers the RLS to provide presence event notification to the watcher. The details of the signalling flows are as follows:

1. **NOTIFY request (PS to S-CSCF) - see example in table A.5.3.1-1**

   The PS determines which authorized watchers are entitled to receive presence information. For each appropriate watcher, the PS sends a NOTIFY request that contains the updated state of presence information. In this example the notification is only sent to the RLS.

   The NOTIFY request may either contain the complete set of presence information, or only those presence information that have changed since the last notification. In this example, the complete set of presence information is sent.

Table A.5.3.1-1: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:rls.home1.net SIP/2.0
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=623551024"; orig-ioi=home12.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>
From: <sip:user2_public1@home2.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: gahjt393yhakfh83hfasl98a
CSeq: 43 NOTIFY
Subscription-State: active;expires=5000
Event: presence
Contact: <sip:ps.home2.net>
Content-Type: application/pidf+xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"

             xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"

             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
        entity="pres:user2_public1@home2.net ">

     <tuple id="a8098a.672364762364">
       <status>
         <basic>open</basic>
       </status>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <rp:class>sip</rp:class>
      <pcp:servcaps>
          <pcp:video>false</pcp:video>
          <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8>sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T17:35:29Z</timestamp>
     </tuple>

     <dm:person id="438">
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>

         <rp:activities><rp:meeting/></rp:activities>
         <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>

     </dm:person>

   </presence>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

2. **NOTIFY request (S-CSCF to RLS) - see example in table A.5.3.1-2**

   The S-CSCF#1 forwards the NOTIFY request to the RLS.

Table A.5.3.1-2: NOTIFY request (S-CSCF to RLS)

```
NOTIFY sip:rls.home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bKehuehjgt, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 69
P-Charging-Vector:
P-Charging-Function-Addresses:
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the RLS.

3. **200 (OK) response (RLS to S-CSCF) - see example in table A.5.3.1-3**

   The RLS generates a 200 (OK) response to the NOTIFY request.

Table A.5.3.1-3: 200 (OK) response (RLS to S-CSCF)

```
SIP/2.0 200 OK
Via:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=623551024"; orig-
      ioi=home1.net:term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

P-Charging-Vector: The RLS stores the terminating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

4. **200 (OK) response (S-CSCF to PS) - see example in table A.5.3.1-4**

   The S-CSCF#1 forwards the 200 (OK) response to the PS.

Table A.5.3.1-4: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=623551024"; orig-
      ioi=home1.net:term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

5. **NOTIFY request (RLS to S-CSCF#1) - see example in table A.5.3.1-5**

The RLS may decide to wait for other notifications and combine them in a single notification towards the UE or it sends the notification to the UE without any waiting. In this example, the RLS does not wait for other notifications.

Table A.5.3.1-5: NOTIFY request (RLS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=723551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_list1@home1.net>;;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: gahjt393yhakfh83hfasl98a
CSeq: 90 NOTIFY
Subscription-State: active;expires=4500
Require: eventlist
Event: presence
Contact: <sip:rls.home1.net>
Content-Type: multipart/related;type="application/rlmi+xml";
       start="<njhhsdhj@rls.home1.net>";boundary="70UBfW7L78hjgfgUPe5z"
Content-Length: (...)

--70UBfW7L78hjgfgUPe5z
Content-Transfer-Encoding: binary
Content-ID: <njhhsdhj@rls.home1.net>
Content-Type: application/rlmi+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
  <list xmlns="urn:ietf:params:xml:ns:rmli"
        uri="sip:user1_list1@home1.net"
                   version="2"
                   fullState="false"
     <resource uri="pres:user2_public1@home2.net">
       <name>Kovacs Janos</name>
       <instance id="hqzsuxtfyq" state="active" cid="uhjgfd@rls.home1.net"/>
     </resource>
  </list>

--70UBfW7L78hjgfgUPe5z
Content-Transfer-Encoding: binary
Content-ID: <uhjgfd@rls.home1.net>
Content-Type: application/pidf+xml;charset="UTF-8"

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
              xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"
             xmlns:dm="urn:ietf:params:xml:ns:pidf:data-model"
             xmlns:pcp="urn:ietf:params:xml:ns:pidf:caps"
             xmlns:ci="urn:ietf:params:xml:ns:pidf:cipid"
             entity="pres:user2_public1@home2.net">

     <tuple id="a8098a.672364762364">
       <status>
```

```
         <basic>open</basic>
       </status>
       <rp:class>sip</rp:class>
       <rp:privacy><text/></rp:privacy>
       <rp:status-icon>http://example.com/~user2/icon.gif</rp:status-icon>
       <pcp:servcaps>
         <pcp:video>false</pcp:video>
         <pcp:audio>true</pcp:audio>
       </pcp:servcaps>
       <contact priority="0.8>sip:user2_public1@home2.net</contact>
       <note xml:lang="en">Don't Disturb Please!</note>
       <note xml:lang="fr">Ne derangez pas, s'il vous plait</note>
       <timestamp>2003-08-27T17:35:29Z</timestamp>
     </tuple>

     <dm:person>
       <rp:class>presentity</rp:class>
       <ci:homepage>http://example.com/~user2</ci:homepage>
       <ci:card>http://example.com/~user2/card.vcd</ci:card>
       <rp:activities><rp:meeting/></rp:activities>
       <rp:place-type until="2003-08-27T17:30:00Z"><rp:office/></rp:place-type>
     </dm:person>

   </presence>

--70UBfW7L78hjgfgUPe5z
```

P-Charging-Vector: The RLS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The RLS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

6. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.5.3.6**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.5.3.1-6: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=723551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

7. **NOTIFY request (P-CSCF to UE) - see example in table A.5.3.1-7**

   The P-CSCF forwards the NOTIFY request to the UE.

Table A.5.3.1-7: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Require:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

8. **200 (OK) response (UE to P-CSCF) - see example in table A.5.3.1-8**

   The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.5.3.1-8: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

9. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.5.3.1-9**

   The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.5.3.1-9: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=723551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

10. **200 (OK) response (S-CSCF to RLS) – see example in table A.5.3.1-10**

    The S-CSCF forwards the response to the RLS in the home network of the presentity.

Table A.5.3.1-10: 200 (OK) response (S-CSCF to RLS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP rls.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=723551024"; orig-
      ioi=home1.net:term-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the RLS.

A.6 PUA subscribing to his own watcher list and receiving notification of new watcher subscriptions

A.6.1 Introduction

This subclause covers the signalling flows that show how a PUA can subscribe to his own watcher list.

A.6.2 PUA subscribing to watcher list and receiving a notification of an already pending watcher subscription followed by a notification of a subscription from a new watcher not already in the watcher list

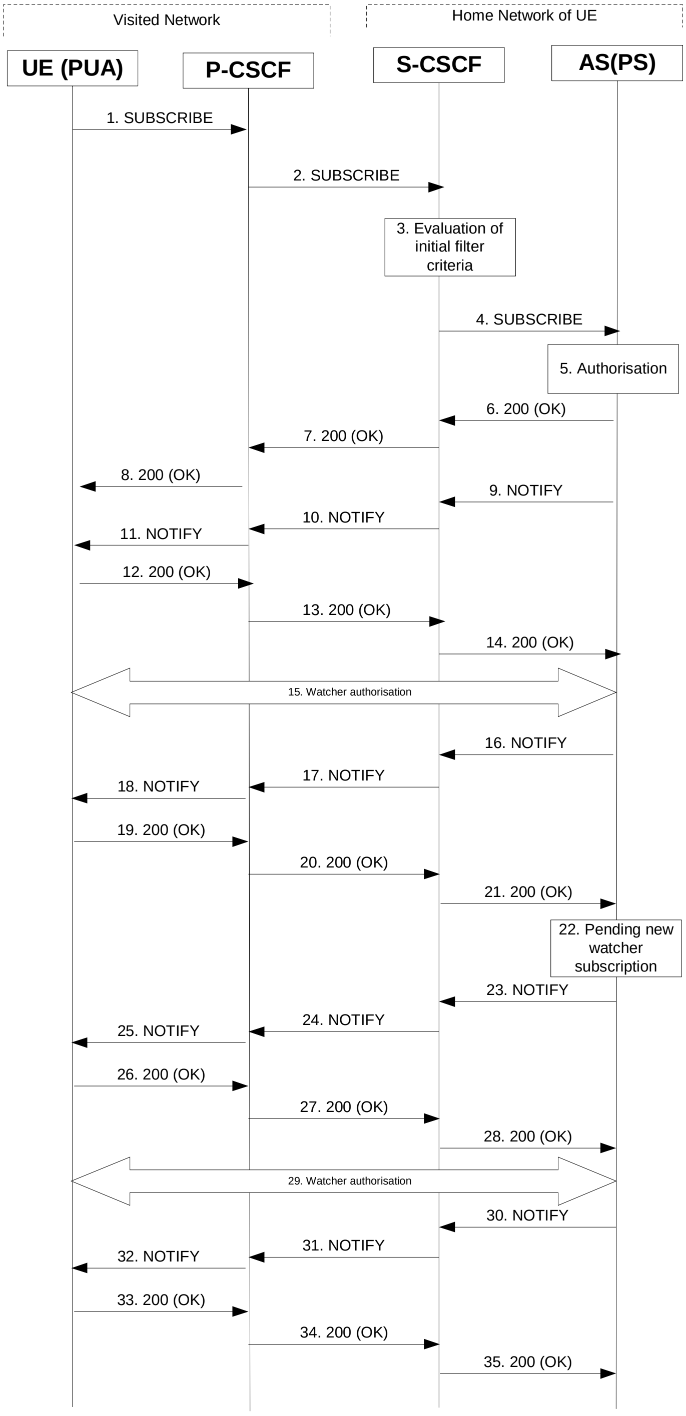

Figure A.6.2-1: PUA subscribing to watcher list and receiving a notification of an already pending watcher subscription followed by a notification of a subscription from a new watcher not already in the watcher list

Figure A.6.2-1 shows a PUA subscribing to the watcher list and receiving a notification of an already pending watcher subscription followed by a notification of a subscription from a new watcher not already in the watcher list. In this example the default watcherinfo subscription filtering policy is applied meaning that a partial state of a watcher-info document is transported in the notifications. The details of the signalling flows as follows:

1. **SUBSCRIBE request (UE to P-CSCF) – see example in table A.6.2-1**

   The presentity wishes to watch his own watcher information, therefore he subscribes for the watcher information template-package of presence. The UE generates a SUBSCRIBE request containing the presence.winfo event, together with an indication of the length of time this periodic subscription should last.

Table A.6.2-1: SUBSCRIBE request (UE to P-CSCF)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Max-Forwards: 70
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
P-Preferred-Identity: <sip:user1_public1@home1.net>
Privacy: none
From: <sip:user1_public1@home1.net>;tag=31415
To: <sip:user1_public1@home1.net>
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 123 SUBSCRIBE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; spi-c=98765432; spi=87654321; port-
      c=8642; port-s=7531
Event: presence.winfo
Expires: 7200
Accept: application/watcherinfo+xml
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp>
Content-Length: 0
```

Request URI: Public user identity whose events the subscriber subscribes to. In this case the Public User Identity of the presentity in SIP URI format.

Event: This field is populated with the value "presence.winfo" to specify the use of the watcher information template-package of presence.

Accept: This field is populated with the value 'application/watcherinfo+xml' indicating that the UE supports this body type for notification.

To: Same as the Request-URI.

2. **SUBSCRIBE request (P-CSCF to S-CSCF) – see example in table A.6.2-2**

   The P-CSCF looks up the serving network information for the public user identity that was stored during the registration procedure. The SUBSCRIBE request is forwarded to the S-CSCF. A Route header is inserted into SUBSCRIBE request.

Table A.6.2-2: SUBSCRIBE request (P-CSCF to S-CSCF)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1 ,SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Access-Network-Info:
Max-Forwards: 69
P-Asserted-Identity: <sip:user1_public1@home1.net>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
Route: <sip:orig@scscf1.home1.net;lr>
Record-Route: <sip:pcscf1.visited1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

3. **Evaluation of initial filter criteria**

   The S-CSCF validates the service profile of this subscriber and evaluates the initial filter criteria. For sip:user1_public1@home1.net the S-CSCF has originating initial Filter Criteria with Service Point Trigger of Method = SUBSCRIBE AND Event = "presence.winfo" that informs the S-CSCF to route the SUBSCRIBE request to the AS sip:ps.home1.net.

4. **SUBSCRIBE request (S-CSCF to PS) – see example in table A.6.2-4**

   The S-CSCF forwards the SUBSCRIBE request to the PS.

Table A.6.2-4: SUBSCRIBE request (S-CSCF to PS)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Access-Network-Info:
Max-Forwards: 68
P-Asserted-Identity: <sip:user1_public1@home1.net>, <tel:+1-212-555-1111>
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Privacy:
Route: <sip:ps.home1.net;lr>, <sip:scscf1.home1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From:
To:
Call-ID:
CSeq:
Event:
Expires:
Accept:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the PS.

5. **Authorization**

The PS performs the necessary authorization checks on the originator. In this example, the originator is the owner of the watcher information, so he/she is authorized to see the full watcher information.

In other examples (when the originator is not the owner of the watcher information) subscribers are only allowed to monitor the state of their own subscription, which means that they will receive notifications only containing the state of their own subscription. This requires that a terminating initial Filter Criteria with Service Point Trigger of Method = SUBSCRIBE AND Event = "presence.winfo" has been defined for the user sip:user1_public1@home1.net.

6. **200 (OK) response (PS to S-CSCF) - see example in table A.6.2-6**

The PS sends the response to the S-CSCF.

Table A.6.2-6: 200 (OK) response (PS to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK344a65.1, SIP/2.0/UDP
      pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"; orig-
      ioi=home1.net:term-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Record-Route:
From:
To: <sip:user1_public1@home1.net>;tag=151170
Call-ID:
CSeq:
Expires:
Contact: <sip:ps.home1.net>
Content-Length: 0
```

P-Charging-Vector: The PS stores the originating Inter Operator Identifier (IOI) parameter and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS stores the P-Charging-Function-Addresses header field and passes this header to the S-CSCF.

7. **200 (OK) response (S-CSCF to P-CSCF) - see example in table A.6.2-7**

The S-CSCF forwards the response to the P-CSCF.

Table A.6.2-7: 200 (OK) response (S-CSCF to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK120f34.1, SIP/2.0/UDP
      [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Record-Route:
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

P-Charging-Vector: The S-CSCF stores the terminating Inter Operator Identifier (IOI) parameter.

8. **200 (OK) response (P-CSCF to UE) - see example in table A.6.2-8**

The P-CSCF forwards the response to the PUA in the UE.

Table A.6.2-8: 200 (OK) response (P-CSCF to UE)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKehuefdam
Record-Route: <sip:orig@scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Expires:
Contact:
Content-Length:
```

9. **NOTIFY request (PS to S-CSCF) - see example in table A.6.2-9**

After the PS generated a 200 (OK) response to the SUBSCRIBE request from the UE, it generates a NOTIFY request containing the current state of the watcher information. The watcher information contains one pending subscription.

Table A.6.2-9 NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1@home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 89 NOTIFY
Subscription-State: active;expires=7200
Event: presence.winfo
Contact: <sip:ps.home1.net>
Content-Type: application/watcherinfo+xml
Content-Length: (...)

<?xml version="1.0"?>
   <watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
                version="0" state="full">
     <watcher-list resource="sip:user1_public1@home1.net" package="presence">
       <watcher id="77ajsyy76" event="subscribe"
                status="pending">sip:user2_public1@home2.net</watcher>
     </watcher-list>
   </watcherinfo>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

10. **NOTIFY request (S-CSCF to P-CSCF) – see example in table A.6.2-10**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.6.2-10: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

11. **NOTIFY request (P-CSCF to UE) - see example in table A.6.2-11**

The P-CSCF forwards the NOTIFY request to the PUA in the UE.

Table A.6.2-11: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
Max-Forwards: 68
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

12. **200 (OK) response (UE to P-CSCF) – see example in table A.6.2-12**

The PUA on the UE determines that this is a full state watcher-info document and replaces any current watcher-info with the new document. The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.6.2-12: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

13. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.6.2-13**

The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.6.2-13: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

14. **200 (OK) response (S-CSCF to PS) – see example in table A.6.2-14**

The P-CSCF forwards the response to the PS in the home network of the UE.

Table A.6.2-14: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=123551024"; orig-
      ioi=home1.net:term-ioi=home1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF inserts the terminating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

15. **Authorization of watcher**

The presentity determines to allow the watcher to access the presence information. The PUA modifies the subscription authorization policy by authorizing presence information for sip:user2_public1@home1.net.

16. **NOTIFY request (PS to S-CSCF) – see example in table A.6.2-16**

The authorization event means changes in the watcher information, which triggers a new NOTIFY request. The watcher information included in the NOTIFY request contains only information on the watcher whose state has changed, which in this example is the accepted subscription of sip:user2_public1@home1.net.

Table A.6.2-16: NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 90 NOTIFY
Subscription-State: active;expires=4900
Event: presence.winfo
Contact: <sip:ps.home1.net>
Content-Type: application/watcherinfo+xml
Content-Length: (...)

<?xml version="1.0"?>
   <watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
                version="0" state="partial">
     <watcher-list resource="sip:user1_public1@home1.net" package="presence">
       <watcher id="77ajsyy76" event="subscribe"
                status="active">sip:user2_public1@home2.net</watcher>
     </watcher-list>
   </watcherinfo>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

17. **NOTIFY request (S-CSCF to P-CSCF) – see example in table A.6.2-17**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.6.2-17: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF passes this header received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

18. **NOTIFY request (P-CSCF to UE) - see example in table A.6.2-18**

The P-CSCF forwards the NOTIFY request to the PUA in the UE.

Table A.6.2-18: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
Max-Forwards: 68
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Contact:
Content-Type:
Content-Length:

(…)
```

19. **200 (OK) response (UE to P-CSCF) - see example in table A.6.2-19**

The PUA determines that this is a full state watcher-info document and replaces any current watcher-info with the new document. The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.6.2-19: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

20. **200 (OK) response (P-CSCF to S-CSCF) – see example in table A.6.2-20**

The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.6.2-20: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

21. **200 (OK) response (S-CSCF to PS) – see example in table A.6.2-21**

The P-CSCF forwards the response to the PS in the home network of the UE.

Table A.6.2-21: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=223551024"; orig-ioi=home1.net
      term-ioi=visited1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The PS inserts the originating Inter Operator Identifier (IOI) parameter received and.populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

22. **Pending new watcher subscription**

The PS receives a SUBSCRIBE request from a new watcher and performs the necessary authorization checks on the originator and determines that this is a new watcher that is not yet in the watcher list.

23. **NOTIFY request (PS to S-CSCF) - see example in table A.6.2-23**

The PS generates a NOTIFY request containing watcher information of the new watcher pending subscription. Thus, the watcher information contains the partial state.

Table A.6.2-23 NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 90 NOTIFY
Subscription-State: active;expires=5000
Event: presence.winfo
Content-Type: application/watcherinfo+xml
Contact: <sip:ps.home1.net;lr>
Content-Length: (...)

<?xml version="1.0"?>
   <watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
                version="0" state="partial">
     <watcher-list resource="sip:user1_public1@home1.net" package="presence">
       <watcher id="34bytzx54" event="subscribe"
                status="pending">sip:user3_public1@home3.net</watcher>
     </watcher-list>
   </watcherinfo>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

24. **NOTIFY request (S-CSCF to P-CSCF) - see example in table A.6.2-24**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.6.2-24: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Content-Type:
Contact:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

25. **NOTIFY request (P-CSCF to UE) - see example in table A.6.2-25**

The P-CSCF forwards the NOTIFY request to the PUA in the UE.

Table A.6.2-25: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Content-Type:
Contact:
Content-Length:

(…)
```

26. **200 (OK) response (UE to P-CSCF) - see example in table A.6.2-26**

The PUA determines that this is a partial state notification of watcher-info and adds the new pending subscription to its existing watcher-info document. The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.6.2-26: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

27. **200 (OK) response (P-CSCF to S-CSCF) - see example in table A.6.2-27**

The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.6.2-27: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"
From:
To:
Call-ID:
CSeq:
Content-Length:
```

28. **200 (OK) response (S-CSCF to PS) - see example in table A.6.2-28**

The P-CSCF forwards the response to the PS in the home network of the UE.

Table A.6.2-28: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=323551024"; orig-ioi=home1.net;
      term-ioi=visited1.net
From:
To:
Call-ID:
CSeq:
Content-Length:
```

P-Charging-Vector: The S-CSCF insertes the originating Inter Operator Identifier (IOI) parameter received and populates the identifier of its own network to the terminating Inter Operator Identifier (IOI) parameter of this header.

29. **Authorization of watcher**

The presentity determines to allow the watcher to access the presence information. The PUA modifies the authorization policy by authorizing presence information for sip:user3_public1@home3.net.

30. **NOTIFY request (PS to S-CSCF) - see example in table A.6.2-30**

The authorization event means changes in the watcher information, which triggers a new NOTIFY request. The watcher information included in the NOTIFY request contains the accepted subscription of sip:user3_public1@home3.net.

Table A.6.2-30 NOTIFY request (PS to S-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 70
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"; orig-ioi=home1.net
P-Charging-Function-Addresses: ccf=[5555::b99:c88:d77:e66]; ccf=[5555::a55:b44:c33:d22];
      ecf=[5555::1ff:2ee:3dd:4ee]; ecf=[5555::6aa:7bb:8cc:9dd]
Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1.home1.net>;tag=31415
Call-ID: b89rjhnedlrfjflslj40a222
CSeq: 90 NOTIFY
Subscription-State: active;expires=4900
Event: presence.winfo
Content-Type: application/watcherinfo+xml
Contact: <sip:ps.home1.net;lr>
Content-Length: (...)

<?xml version="1.0"?>
   <watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
                version="0" state="partial">
     <watcher-list resource="sip:user1_public1@home1.net" package="presence">
       <watcher id="34bytzx54" event="subscribe"
                status="active">sip:user3_public1@home3.net</watcher>
     </watcher-list>
   </watcherinfo>
```

P-Charging-Vector: The PS populates the icid parameter with a globally unique value and populates the identifier of its own network to the originating Inter Operator Identifier (IOI) parameter of this header.

P-Charging-Function-Addresses: The PS populates the P-Charging-Function-Addresses header field to be passed to the S-CSCF.

31. **NOTIFY request (S-CSCF to P-CSCF) – see example in table A.6.2-31**

The S-CSCF forwards the NOTIFY request to the P-CSCF.

Table A.6.2-31: NOTIFY request (S-CSCF to P-CSCF)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
Max-Forwards: 69
P-Charging-Vector: icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=423551024"
P-Charging-Function-Addresses:
Route: <sip:pcscf1.visited1.net;lr>
Record-Route: <sip:scscf1.home1.net;lr>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Content-Type:
Contact:
Content-Length:

(…)
```

P-Charging-Vector: The S-CSCF stores the originating Inter Operator Identifier (IOI) parameter received.

P-Charging-Function-Addresses: The S-CSCF stores the P-Charging-Function-Addresses header field and passes this header to the P-CSCF.

32. **NOTIFY request (P-CSCF to UE) - see example in table A.6.2-32**

    The P-CSCF forwards the NOTIFY request to the PUA in the UE.

Table A.6.2-32: NOTIFY request (P-CSCF to UE)

```
NOTIFY sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf1.home1.net;branch=z9hG4bK351g45.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
Max-Forwards: 68
Record-Route: <sip:scscf1.home1.net;lr>, <sip:pcscf1.home1.net:7531;lr;comp=sigcomp>
From:
To:
Call-ID:
CSeq:
Subscription-State:
Event:
Content-Type:
Contact:
Content-Length:

(…)
```

33. **200 (OK) response (UE to P-CSCF) - see example in table A.6.2-33**

    The PUA determines that this is a partial state notification of watcher-info and updates the active subscription to its existing watcher-info document. The UE acknowledges the NOTIFY request with a 200 (OK) response to the P-CSCF.

Table A.6.2-33: 200 (OK) response (UE to P-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf1.visited1.net;branch=240f34.1, SIP/2.0/UDP
      scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP ps.home2.net;branch=z9hG4bK348923.1
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
From:
To:
Call-ID:
CSeq:
Content-Length: 0
```

34. **200 (OK) response (P-CSCF to S-CSCF) - see example in table A.6.2-34**

    The P-CSCF forwards the 200 (OK) response to the S-CSCF.

Table A.6.2-34: 200 (OK) response (P-CSCF to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
      ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
From:
To:
Call-ID:
CSeq:
Content-Length:
```

35. **200 (OK) response (S-CSCF to PS) - see example in table A.6.2-35**

The P-CSCF forwards the response to the PS in the home network of the UE.

Table A.6.2-35: 200 (OK) response (S-CSCF to PS)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ps.home1.net;branch=z9hG4bK240f34.1
P-Access-Network-Info:
From:
To:
Call-ID:
CSeq:
Content-Length:
```

A.7 PNA subscription for the reg-event package

Figure A.7-1 shows the registration signalling flow for the scenario when the user is not registered. For the purpose of this registration signalling flow, the subscriber is considered to be roaming. This signalling flow also shows the authentication of the private user identity.

This is followed by the subscription procedure for the reg-event package, whereby the PNA requests to be notified by the S-CSCF when a registration event has occurred. This is done using the 'reg-event' package as described in 3GPP TS 24.229 [9].

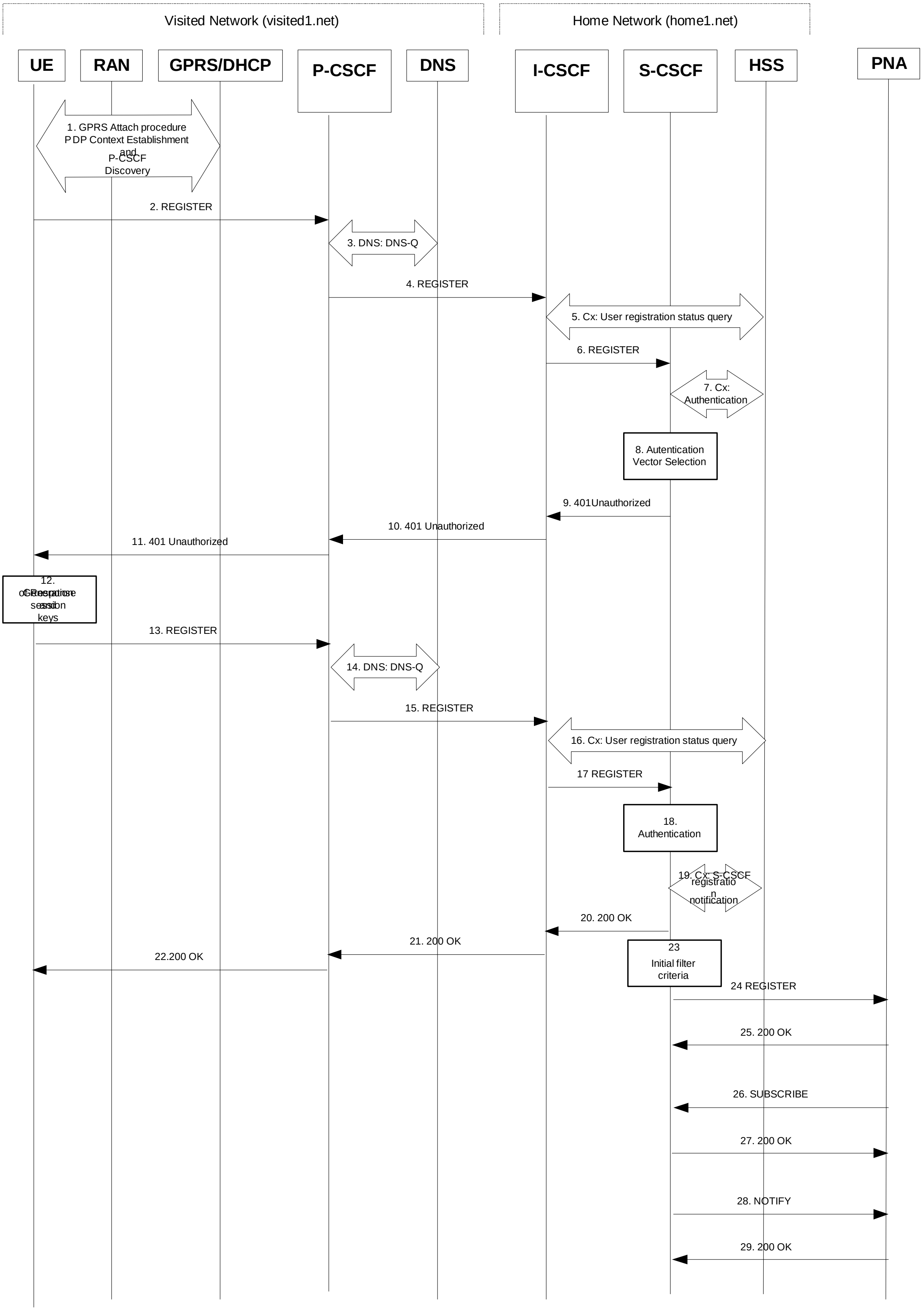

Figure A.7-1: Registration signalling: user not registered

1-22. **See 3GPP TS 24.228 [8], subclause 6.2 steps 1 through 22**

23. **Initial filter criteria**

The S-CSCF analyses the incoming request against the initial filter criteria and decides to send a third-party REGISTER request to the PNA.

24. **REGISTER request (S-CSCF to PNA) - see example in table A.7-24**

This signalling flow forwards the REGISTER request from the S-CSCF to the PNA.

Table A.7-24: REGISTER request (S-CSCF to PNA)

```
REGISTER sip:ps.home1.net SIP/2.0
Via: SIP/2.0/UDP sip:scscf1.home1.net
Max-Forwards: 70
P-Access-Network-Info:
P-Visited-Network-ID:
P-Charging-Vector:
P-Charging-Function-Addresses:
From: sip:scscf1.home1.net
To: <sip:user1_public1@home1.net>
Contact: <sip:scscf1.home1.net>
Expires: 600000
Call-ID: apb03a0s09dkjdfglkj49112
CSeq: 43 REGISTER
Content-Length: 0
```

25. **200 OK response (PNA to S-CSCF) - see example in table A.7-25**

The PNA sends a 200 (OK) response to the S-CSCF indicating that Registration was successful.

Table A.7-25: 200 OK response (PNA to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP sip:scscf1.home1.net
From:
To:
Call-ID:
Contact: <sip:scscf1.home1.net>;expires=600000
CSeq:
Date: Wed, 11 July 2001 08:49:37 GMT
Content-Length:
```

26. **SUBSCRIBE request (PNA to S-CSCF) - see example in table A.7-26**

The PNA sends the SUBSCRIBE request for the reg event package.

Table A.7-26: SUBSCRIBE request (PNA to S-CSCF)

```
SUBSCRIBE sip:user1_public1@home1.net SIP/2.0
Via: SIP/2.0/UDP sip:ps.home1.net
Max-Forwards: 70
P-Asserted-Identity: <sip:ps.home1.net>
Privacy: none
From: <sip:ps.home1.net>;tag=31415
To: <sip:user1_public1@home1.net>
Call-ID: dre36d2v32gnlgiiomm72445
CSeq: 61 SUBSCRIBE
Event: reg
Expires: 600000
Accept: application/reginfo+xml
Contact: <sip:ps.home1.net>
Content-Length: 0
```

From: This header is populated with the SIP URI that identifies the PNA.

Contact: This is where the NOTIFY requests for this subscription will be sent.

Event: This field is set to the value 'reg' to specify the use of the reg event package.

Accept: This field is set to the value "application/reginfo+xml".

27. **200 (OK) response (S-CSCF to PNA) - see example in table A.7-27**

The S-CSCF sends a 200 (OK) response to the PNA indicating that registration was successful.

Table A.7-27: 200 (OK) response (S-CSCF to PNA)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP sip:ps.home1.net
P-Asserted-Identity: <sip:scscf1.home1.net>
Privacy:
From:
To: <sip:user1_public1@home1.net>;tag=151170
Call-ID:
CSeq:
Contact: <sip:scscf1.home1.net>
Expires:
Content-Length:
```

Expires: If value of the Expires header in SUBSCRIBE request is different from the one received in REGISTER method, then the value of Expires header in the 200 (OK) response is set to match the value of Expires header in REGISTER method.

28. **NOTIFY request (S-CSCF to PNA) - see example in table A.7-28**

The S-CSCF sends a first NOTIFY request towards the PNA in order to inform the PNA about the registration status of monitored user.

Table A.7-28: NOTIFY request (S-CSCF to PNA)

```
NOTIFY sip:pcscf1.visited1.net SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1
Max-Forwards: 70
From: <sip:user1_public1@home1.net>;tag=151170
To: <sip:user1_public1@pcscf1.visited1.net>;tag=31415
Call-ID: dre36d2v32gnlgiiomm72445
CSeq: 42 NOTIFY
Subscription-State: active;expires=600000
Event: reg
Content-Type: application/reginfo+xml
Contact: <sip:scscf1.home1.net>
Content-Length: (...)

<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo"
                version="1" state="full">
     <registration aor="sip:user1_public1@home1.net" id="a7" state="active">
         <contact id="76" state="active" event="registered">
             <uri>sip:[5555::aaa:bbb:ccc:ddd]</uri>
         </contact>
     </registration>
     <registration aor="sip:user1_public2@home1.net" id="a8" state="active">
         <contact id="77" state="active" event="created">
             <uri>sip:[5555::aaa:bbb:ccc:ddd]</uri>
         </contact>
     </registration>
     <registration aor="tel:+358504821437" id="a9" state="active">
         <contact id="78" state="active" event="created">
             <uri>sip:[5555::aaa:bbb:ccc:ddd]</uri>
         </contact>
     </registration>
</reginfo>
```

From: The tag of this field matches that of the To; field in the received 200 (OK) response for the SUBSCRIBE request.

Content-Type: Set to the value of the Accept header received in the SUBSCRIBE request or "application/reginfo+xml" if the Accept header was not present in the SUBSCRIBE request.

The message body in the NOTIFY request that carries the subscriber's registration state is formed as indicated in 3GPP TS 24.229 [9].

29. **200 (OK) response (PNA to S-CSCF) - see example in table A.7-29**

PNA sends the 200 (OK) response to the S-CSCF.

Table A.7-29: 200 (OK) response (PNA to S-CSCF)

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1
From:
To:
Call-ID:
CSeq:
Content-Type:
Content-Length: 0
```

A.8 Example signalling flows of HTTP based presence service operation

A.8.1 Introduction

This subclause shows signalling flows relating to the manipulation of presence service data over the Ut reference point using XCAP. The flows only shows the signalling between the XCAP server and the XCAP client, thus possible proxies located in between the entities are not shown in the example signalling flows.

Each example signalling flow shows several sequences of manipulation of data for the presence service.

NOTE: Error conditions are not considered in the examples e.g. if authorization checks fail in the XCAP server, XML Schema compliancy checks fail or the file specified by the URI does not exist then an appropriate 4xx response is sent to the client.

Clarifications how XCAP is using HTTP are described in draft-ietf-simple-xcap [33].

NOTE: The authentication proxy resides between UE (XCAP client) and AS (XCAP server), and examples of signalling flows for the authentication proxy are provided in 3GPP TS 24.109 [7].

A.8.2 Signalling flows demonstrating how XCAP clients manipulate resource lists

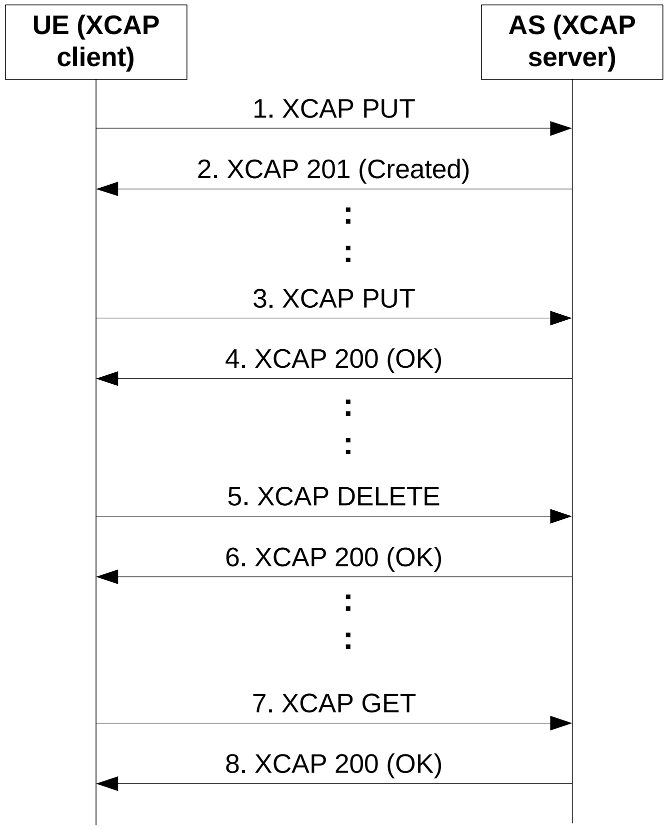

Figure A.8.2-1: XCAP client manipulating a resource list on XCAP server

Figure A.8.2-1 shows a how a XCAP client may manipulate a resource list on a XCAP server. The details of the signalling flows are as follows:

1. **XCAP PUT request (XCAP client to XCAP server - see example in table A.8.2-1**

   The XCAP client generates an XCAP PUT request to create a new resource list on the XCAP server. The resource list has one entry.

Table A.8.2-1: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.home1.net/services/resource-lists/users/user1/pf.xml HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:17 GMT
Content-Type: application/resource-lists+xml
Content-Length: (…)

<?xml version="1.0" encoding="UTF-8"?>
   <resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists">
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="urn:ietf:params:xml:ns:resource-lists">
     <list name="Presence_fellows">
       <entry uri="sip:user2_public1@home2.net">
         <display-name>User2</display-name>
       </entry>
     </list>
   </resource-lists>
```

2. **XCAP 201 (Created) response (XCAP server to XCAP client) – see example in table A.8.2-2**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to create a file, the XCAP server sends an XCAP 201 (Created) response to the XCAP client.

Table A.8.2-2: XCAP 201 (Created) response (XCAP server to XCAP client)

```
HTTP/1.1 201 Created
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "aaa"
Date: Thu, 08 Jan 2004 10:50:35 GMT
Content-Length: 0
```

3. **XCAP PUT request (XCAP client to XCAP server) – see example in table A.8.2-3**

   The XCAP client adds a new entry to the previously created resource list by generating a new XCAP PUT request.

Table A.8.2-3: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.home1.net/services/resource-lists/users/user1/pf.xml/~~/resource-
      lists/list%5b@name=%22Presence_fellows%22%5d/entry HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:17 GMT
Content-Type: application/xcap-el
Content-Length: (…)

         <entry uri="sip:user3_public1@home3.net">
           <display-name>User3</display-name>
         </entry>
```

4. **XCAP 200 (OK) response (XCAP server to XCAP client) - see example in table A.8.2-4**

   After the XCAP server has performed the necessary authorization checks, XML document validations and XML schema compliancy checks the XCAP server sends an XCAP 200 (OK) response to the XCAP client.

Table A.8.2-4: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "aab"
Date: Thu, 08 Jan 2004 10:50:45 GMT
Content-Length: 0
```

5. **XCAP DELETE request (XCAP client to XCAP server) - see example in table A.8.2-5**

   The XCAP client decides to delete the entry "user2" from the resource list. The XCAP client generates an XCAP DELETE request.

Table A.8.2-5: XCAP DELETE request (XCAP client to XCAP server)

```
DELETE http://xcap.home1.net/services/resource-lists/users/user1/pf.xml/~~/resource-
      lists/list%5b@name=%22Presence_fellows%22%5d/entry%5b@name=user2"%5d HTTP/1.1
Host: oper.example.com:9999
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:14:17 GMT
```

6. **XCAP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.2-6**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to delete an entry from the resource list, the XCAP server sends an XCAP 200 (OK) response.

Table A.8.2-6: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Date: Thu, 08 Jan 2004 11:00:47 GMT
Content-Length: 0
```

7. **XCAP GET request (XCAP client to XCAP server) – see example in table A.8.2-7**

   The XCAP client wishes to check the result of the previous transaction by generating an XCAP GET request.

Table A.8.2-7: XCAP GET request (XCAP client to XCAP server)

```
GET http://xcap.home1.net/services/resource-lists/users/user1/pf.xml HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 11:13:17 GMT
Content-Length: 0
```

8. **XCAP 200 (OK) response (XCAP server to XCAP client) - see example in table A.8.2-8**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to fetch the resource list, the XCAP server sends an XCAP 200 (OK) response to the XCAP client including the resource list in the body of the response.

Table A.8.2-8: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "askdajdsaj"
Date: Thu, 08 Jan 2004 11:50:35 GMT
Content-Type:application/resource-lists+xml
Content-Length: (…)

<?xml version="1.0" encoding="UTF-8"?>
   <resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists">
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="urn:ietf:params:xml:ns:resource-lists">
     <list name="Presence_fellows">
       <entry uri="sip:user3_public1@home3.net">
         <display-name>User3</display-name>
       </entry>
     </list>
   </resource-lists>
```

A.8.3 Signalling flows demonstrating how XCAP clients manipulate presence authorization policy

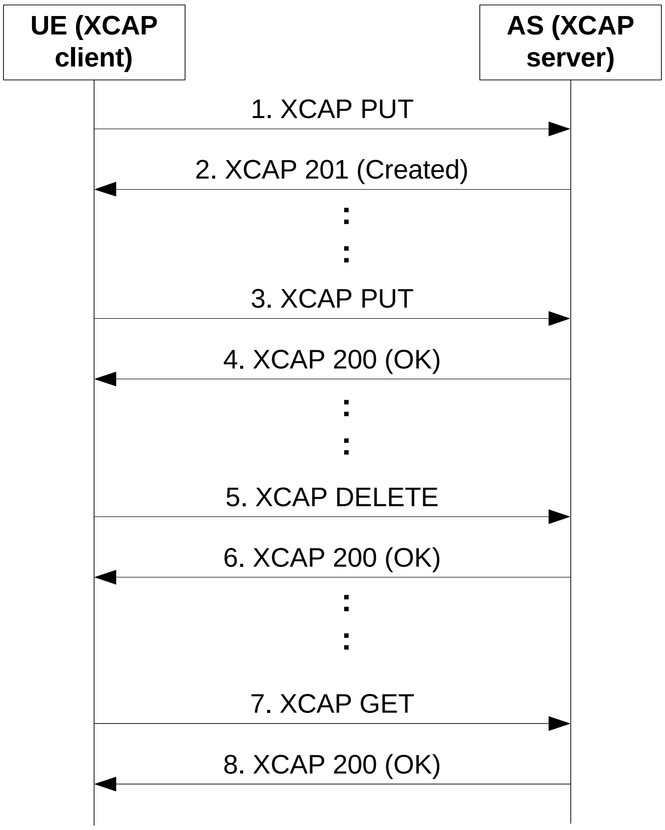

Figure A.8.3-1: XCAP client manipulating presence authorization policy on XCAP server

Figure A.8.3-1 shows a XCAP client manipulating presence authorization policy on a XCAP server. The details of the signalling flows are as follows:

1. **XCAP PUT request (XCAP client to XCAP server) – see example in table A.8.3-1**

   The XCAP client generates an XCAP PUT request to create a presence authorization policy on the XCAP server. The presence authorization policy has one rule allowing for sip:user2_public1@home2.net to see all service information along with the service related servcaps elements defined in draft-ietf-simple-prescaps-ext [25].

Table A.8.3-1: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.home1.net/services/pres-rules/users/user1/ps.xml HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:17 GMT
Content-Type:application/auth-policy+xml
Content-Length: (…)

<?xml version="1.0" encoding="UTF-8"?>
   <ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
     xmlns:sc="urn:ietf:params:xml:ns:pidf:caps"
     xmlns:pr="urn:ietf:params:xml:ns:pres-rules"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="urn:ietf:params:xml:ns:pres-rules">
     <rule id="dsafa43232">
       <conditions>
         <identity>
         <id entity="user2_public1@home2.net"/>
         </identity>
       </conditions>
       <actions>
       <pr:sub-handling>allow</pr:sub-handling>
       </actions>
       <transformations>
         <pr:provide-services>
           <pr:all-services/>
         </pr:provide-services>
           <pr:provide-persons><pr:all-persons/></pr:provide-persons>
           <pr:provide-unknown-attribute name="sc:servcaps">true</pr:provide-unknown-
              attribute>
       </transformations>

     </rule>
   </ruleset>
```

2. **XCAP 201 (Created) response (XCAP server to XCAP client) - see example in table A.8.3-2**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to create a file, the XCAP server sends an XCAP 201 (Created) response to the XCAP client.

Table A.8.3-2: XCAP 201 (Created) response (XCAP server to XCAP client)

```
HTTP/1.1 201 Created
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "bbb"
Date: Thu, 08 Jan 2004 10:50:35 GMT
Content-Length: 0
```

3. **XCAP PUT request (XCAP client to XCAP server) – see example in table A.8.3-3**

   The XCAP client adds a new rule to the previously created presence authorization policy by generating a new XCAP request. The new rule blocks the user named sip:user3_public1@home3.net to see presence information.

Table A.8.3-3: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.home1.net/services/pres-rules/users/user1/ps.xml/~~/permission-
      statements/ruleset/rule%5b2%5d HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:27 GMT
Content-Type: application/xcap-el
Content-Length: (…)

     <rule id="fdsjkf">
       <conditions>
         <identity>
           <id entity="user3_public1@home2.net"/>
         </identity>
       <conditions>
       <actions>
         <pr:sub-handling>block</pr:sub-handling>
       </actions>
```

4. **XCAP 200 (OK) response (XCAP server to XCAP client) - see example in table A.8.3-4**

   After the XCAP server has performed the necessary authorization checks, XML document validations and XML schema compliancy checks the XCAP server sends an XCAP 200 (OK) response to the XCAP client.

Table A.8.3-4: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "bbb2"
Date: Thu, 08 Jan 2004 10:50:45 GMT
Content-Length: 0
```

5. **XCAP DELETE request (XCAP client to XCAP server) - see example in table A.8.3-5**

   The XCAP client decides to delete the rule for sip:user2_public1@home2.net from the authorization policy. The XCAP client generates an XCAP DELETE request.

Table A.8.3-5: XCAP DELETE request (XCAP client to XCAP server)

```
DELETE http://xcap.home1.net/services/pres-
      rules/users/user1/ps.xml/~~/ruleset/rule/statement[@id="dsafa43232"] HTTP/1.1
Host: oper.example.com:9999
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:14:17 GMT
```

6. **XCAP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.3-6**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to delete an entry from the resource list, the XCAP server sends an XCAP 200 (OK) response.

Table A.8.3-6: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Date: Thu, 08 Jan 2004 11:00:47 GMT
Content-Length: 0
```

7. **XCAP GET request (XCAP client to XCAP server) – see example in table A.8.3-7**

   The XCAP client wishes to check the result of the previous transaction by generating an XCAP GET request.

Table A.8.3-7: XCAP GET request (XCAP client to XCAP server)

```
GET http://xcap.home1.net/services/pres-rules/users/user1/ps.xml HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 11:13:17 GMT
Content-Length: 0
```

8. **XCAP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.3-8**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to fetch the resource list, the XCAP server sends an XCAP 200 (OK) response to the XCAP client including the authorization rules in the body of the response.

Table A.8.3-8: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "eiuuekksks"
Date: Thu, 08 Jan 2004 11:50:35 GMT
Content-Type:application/auth-policy+xml
Content-Length: (…)

<?xml version="1.0" encoding="UTF-8"?>
   <ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
     xmlns:sc="urn:ietf:params:xml:ns:pidf:caps"
     xmlns:pr="urn:ietf:params:xml:ns:pres-rules"
     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="urn:ietf:params:xml:ns:pres-rules">
     <rule id="fdsjkf">
       <conditions>
         <identity>
           <id entity="user3_public1@home2.net"/>
         </identity>
       <conditions>
       <actions>
         <pr:sub-handling>block</pr:sub-handling>
       </actions>
```

```
    </ruleset>
```

A.8.4 Storing external content (successful operation)

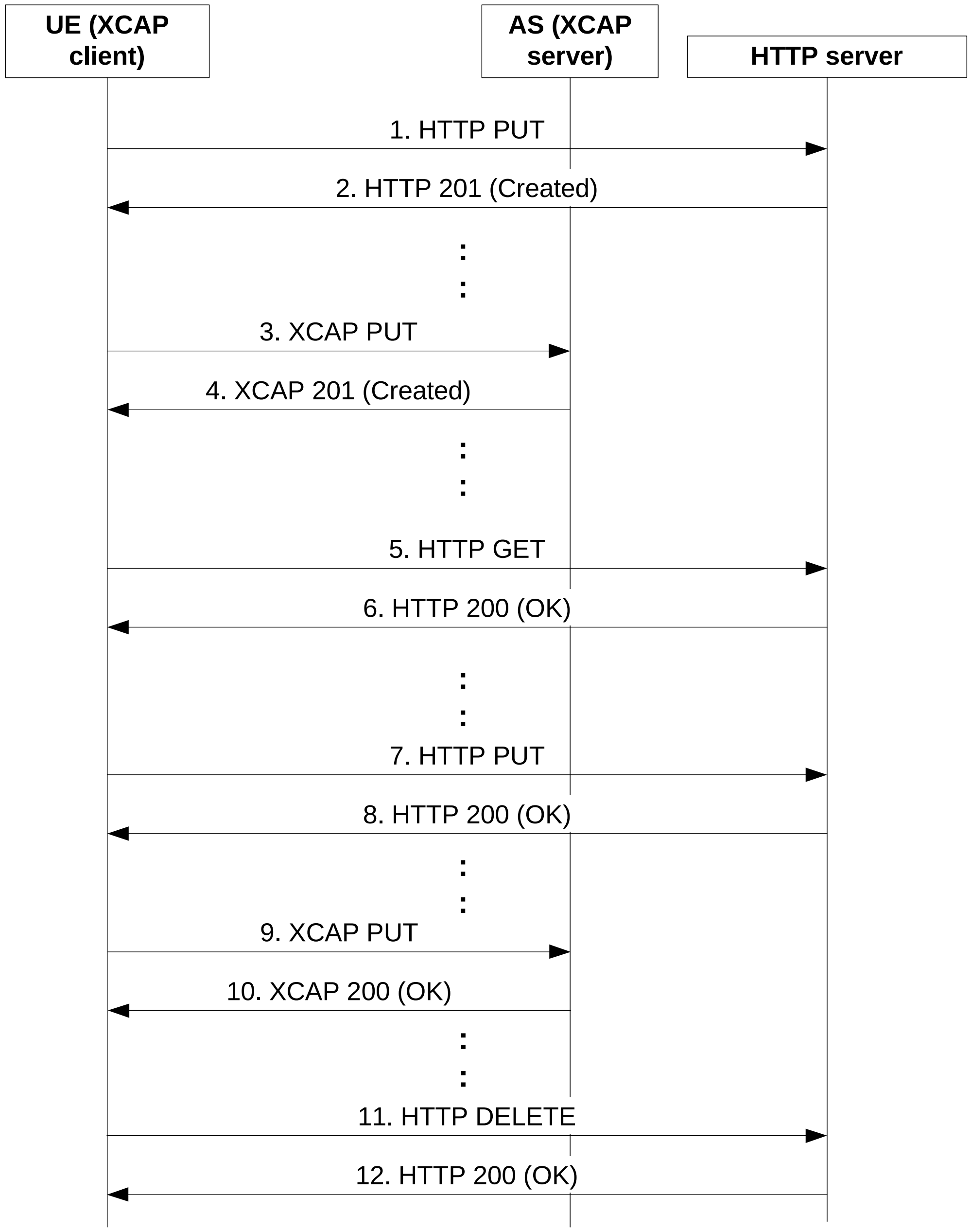

Figure A.8.4.-1: XCAP client manipulating hard-state presence document on XCAP server

Figure A.8.4-1 shows a XCAP client manipulating hard-state presence document on a XCAP server when the presence document has an aggregated storing MIME object with the "application/pidf+xml" content type and any of its extensions. The details of the signalling flows are as follows:

1. **HTTP PUT request (XCAP client to XCAP server) – see example in table A.8.2-1**

   In order to store the content, the XCAP client generates an HTTP PUT request containing the MIME object in the body of the request. The request-URI points to the directory where the content is stored and shows the name of the file to be created.

Table A.8.4-1: HTTP PUT request (XCAP client to XCAP server)

```
PUT http://operator.example.com/services/users/bill/pictureX HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:17 GMT
Content-Type: image/jpeg
Content-Length: (…)

{pictureX.jpg}
```

2. **HTTP 201 (Created) response (XCAP server to XCAP client) – see example in table A.8.4-2**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to create a file the HTTP server sends an HTTP 201 (Created) response to the client.

Table A.8.4-2: HTTP 201 (Created) response (XCAP server to XCAP client)

```
HTTP/1.1 201 Created
Server: Apache/1.3.22 (Unix) mod_perl/1.27Content-Type: text/html
Date: Thu, 08 Jan 2004 10:50:35 GMT
Content-Length: 0
```

3. **XCAP PUT request (XCAP client to XCAP server) - see example in table A.8.2-3**

   The XCAP client generates an XCAP PUT request in order to store XML encoded presence document which includes a URI reference to the MIME object stored on the XCAP server. The AUID part of the request URI is 'pidf-manipulation' as defined in draft-ietf-simple-xcap-pidf-manipulation-usage [34].

Table A.8.4-3: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.example.com/services/pidf-manipulation/users/bill/pidf.xml HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:13:27 GMT
Content-Type: application/pidf+xml
Content-Length: (…)

<?xml version="1.0" encoding="UTF-8"?>
   <presence xmlns="urn:ietf:params:xml:ns:pidf"
          xmlns:rp="urn:ietf:params:xml:ns:pidf:rpid"
          xmlns:ext="urn:ietf:params:xml:ns:ext-cont"
          xmlns:p="urn:ietf.params:xml:ns:pidf:data-model"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="urn:ietf:params:xml:ns:resource-lists"
          entity="sip:bill@example.com">

     <tuple id="123sd">
       <status>
         <basic>open</basic>
       </status>
         <contact>sip:bill@example.com</contact>
     </tuple>

     <p:person>
         <rp:activities><rp:vacation/></rp:activities>
       <ext:photo>
         http://operator.example.com/services/users/bill/pictureX.jpg
       </ext:photo>
     </p:person>
   </presence>
```

4. **XCAP 201 (Created) response (XCAP server to XCAP client) - see example in table A.8.4-4**

   After the XCAP server has performed the necessary authorization checks, XML document validations and XML schema compliancy checks the XCAP server sends an XCAP 201 (Created) response to the XCAP client.

Table A.8.4-4: XCAP 201 (Created) response (XCAP server to XCAP client)

```
HTTP/1.1 201 Created
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "ccc1"
Date: Thu, 08 Jan 2004 10:50:45 GMT
Content-Length: 0
```

5. **HTTP GET request (XCAP client to XCAP server) – see example in table A.8.4-5**

   The XCAP client wishes to fetch the MIME object from the XCAP server. The client generates an HTTP GET request. The request URI points to the directory where the object is stored and indicates the name of the file to be fetched.

Table A.8.4-5: HTTP GET request (XCAP client to XCAP server)

```
GET http://operator.example.com/services/users/bill/pictureX HTTP/1.1
Host: oper.example.com:9999
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 10:43:17 GMT
Accept: image/jpeg
Content-Length: 0
```

6. **HTTP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.4-6**

   After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to fetch the file the XCAP server sends an HTTP 200 (OK) response having the object in the body to the XCAP client.

Table A.8.4-6: HTTP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Date: Thu, 08 Jan 2004 11:00:47 GMT
Content-Type: image/jpeg
Content-Length: (…)

{pictureX}
```

7. **HTTP PUT request (XCAP client to XCAP server) – see example in table A.8.4-7**

   The XCAP client wishes to modify the earlier stored MIME object by replacing the picture X with a new picture X with new content. To modify the object the XCAP client generates an HTTP PUT request using the same request URI as has been used for the modified (old) object. The new object is conveyed in the body of the request.

Table A.8.4-7: HTTP PUT request (XCAP client to XCAP server)

```
PUT http://operator.example.com/services/users/bill/pictureX HTTP/1.1
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 11:13:17 GMT
Content-Type: image/jpeg
Content-Length: (…)

{pictureX.jpg}
```

8. **HTTP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.4-8**

After the XCAP server has performed the necessary authorization checks on the originator to ensure the XCAP client is allowed to replace the existing MIME object with the new one the XCAP server sends an HTTP 200 (OK) response to the XCAP client.

Table A.8.4-8: HTTP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Date: Thu, 08 Jan 2004 11:50:35 GMT
Content-Length: 0
```

9. **XCAP PUT request (XCAP client to XCAP server) – see example in table A.8.4-9**

The XCAP client wishes to remove the MIME object from his presence information. The XCAP client generates an XCAP PUT request to modify the XML encoded presence document to remove the reference to the MIME object from the presence document. The request URI contains a node selector to the requested tuple according to draft-ietf-simple-xcap [33].

Table A.8.4-9: XCAP PUT request (XCAP client to XCAP server)

```
PUT http://xcap.example.com/services/pidf-manipulation/users/bill/pidf.xml/~~/presence/person
  HTTP/1.1
Date: Thu, 08 Jan 2004 11:13:37 GMT
If-Match: "ccc1"
Content-Type: application/xcap-el
Content-Length: (…)

   <p:person>

       <rp:activities><rp:vacation/></rp:activities>

   </ p:person>
```

10. **XCAP 200 (OK) response (XCAP server to XCAP client) - see example in table A.8.4-10**

After the XCAP server has performed the necessary authorization checks, XML document validations and XML Schema compliancy checks the XCAP server sends an XCAP 200 (OK) response to the XCAP client.

Table A.8.4-10: XCAP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Etag: "ccc2"
Date: Thu, 08 Jan 2004 11:50:59 GMT
Content-Length: 0
```

11. **HTTP DELETE request (XCAP client to XCAP server) – see example in table A.8.4-11**

The XCAP client removes the MIME object from the XCAP server by generating an HTTP DELETE request.

Table A.8.4-11: HTTP DELETE request (XCAP client to XCAP server)

```
DELETE http://operator.example.com/services/users/bill/pictureX HTTP/1.1
Host: oper.example.com:9999
User-Agent: IMS subscriber
Date: Thu, 08 Jan 2004 11:52:00 GMT
```

12. **HTTP 200 (OK) response (XCAP server to XCAP client) – see example in table A.8.4-12**

    After the XCAP server has performed the necessary authorization checks on the originator to ensure that the XCAP client is allowed to delete the object, the XCAP server sends an HTTP 200 (OK) response to the XCAP client.

Table A.8.4-12: HTTP 200 (OK) response (XCAP server to XCAP client)

```
HTTP/1.1 200 OK
Server: Apache/1.3.22 (Unix) mod_perl/1.27
Date: Thu, 08 Jan 2004 11:52:35 GMT
Content-Length: 0
```

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
2003-06					Version 0.0.1: Preliminary discussion with editor		
2003-06					Version 0.0.2: Results of preliminary discussion with interested parties		
2003-08	CN1#31	N1-031176			Version 0.0.3: Revised as a result of conference call and email discussion with interested parties		
2003-08	CN1#31	N1-031324			Version 0.0.4: Revised as a result of offline discussions at CN1#31		
2003-09	CN1#31	N1-031365			Version 0.1.0: Revised to included allocated TS number and title as amended by MCC.		
2003-11	CN1#32	N1-03aaaa			Version 0.2.0: Revised to include agreements made in N1-031366.		
2004-05	CN1#34	NP-04xxxx			Version 0.3.0. Revised to include agreements made in N1-040765, N1-040794, N1-040795, N1-040866, N1-040939, N1-040945, N1-040946, N1-040997, N1-040998, N1-040999, N1-041002, N1-041006, N1-041008, N1-041092, N1-041093.		
2004-05	CN1#34	NP-040200			Version 2.0.0 identical to version 0.3.0 for presentation for approval to plenary.		
2004-06	CN#24	NP-040200			Approved by the plenary.	2.0.0	6.0.0
2004-09	CN#25	NP-040426	3	2	Editorial issues	6.0.0	6.1.0
2004-09	CN#25	NP-040380	4	1	Watcher cleanup and alignment with PUA	6.0.0	6.1.0
2004-09	CN#25	NP-040380	5	3	PUA clause restructuring	6.0.0	6.1.0
2004-09	CN#25	NP-040427	6	2	GAA impacts	6.0.0	6.1.0
2004-09	CN#25	NP-040380	7	1	XCAP roles	6.0.0	6.1.0
2004-09	CN#25	NP-040380	8	1	XCAP Change	6.0.0	6.1.0
2004-09	CN#25	NP-040380	9		Policy Capability References	6.0.0	6.1.0
2004-09	CN#25	NP-040380	11		Filter criteria update	6.0.0	6.1.0
2004-09	CN#25	NP-040380	14	1	Enhanced partial publication description	6.0.0	6.1.0
2004-09	CN#25	NP-040380	15	1	Publication Rate Limiting	6.0.0	6.1.0
2004-09	CN#25	NP-040380	17	1	Correction to processing PUBLISH with the "multipart/related" content type	6.0.0	6.1.0
2004-09	CN#25	NP-040380	18		XML document corrections of message flows	6.0.0	6.1.0
2004-12	CN#26	NP-040503	19	1	Clarifications to Ut	6.1.0	6.2.0
2004-12	CN#26	NP-040503	20		Alignment between PUA and watcher for draft-ietf-geopriv-pidf-lo-01	6.1.0	6.2.0
2004-12	CN#26	NP-040599	21	2	Introduction of XCAP client and XCAP server	6.1.0	6.2.0
2004-12	CN#26	NP-040503	22	1	Correction XCAP change flow	6.1.0	6.2.0
2004-12	CN#26	NP-040503	23		Delete Authentication Proxy Requirements	6.1.0	6.2.0
2004-12	CN#26	NP-040503	24	1	Aligning Presence data model with IETF	6.1.0	6.2.0
2004-12	CN#26	NP-040503	25	1	IETF reference update (SIP specific parts)	6.1.0	6.2.0
2004-12	CN#26	NP-040503	26	1	IETF reference update (XCAP)	6.1.0	6.2.0
2004-12	CN#26	NP-040503	27	1	Updates to Partial publication	6.1.0	6.2.0
2004-12	CN#26	NP-040503	28	1	Correction to Watcher Information message flow	6.1.0	6.2.0
2004-12	CN#26	NP-040503	30	1	Preventing loop in RLS subscriptions	6.1.0	6.2.0

2004-12	CN#26	NP-040503	32		Filter criteria update	6.1.0	6.2.0
2005-03	CN#27	NP-050073	34	1	Resolution of references to 24.228	6.2.0	6.3.0
2005-03	CN#27	NP-050078	35		Authentication proxy for presence	6.2.0	6.3.0
2005-03	CN#27	NP-050078	37		XCAP-change clarrification	6.2.0	6.3.0
2005-03	CN#27	NP-050078	38		XCAP-change correction	6.2.0	6.3.0
2005-03	CN#27	NP-050078	39		IFC corrections	6.2.0	6.3.0
2005-06	CP#28	CP-050063	41		SPI to SPT	6.3.0	6.4.0
2005-06	CP#28	CP-050063	43		Reference update: event-list	6.3.0	6.4.0
2005-06	CP#28	CP-050063	44		Reference update: filter	6.3.0	6.4.0
2005-06	CP#28	CP-050063	45		Reference update: xcap	6.3.0	6.4.0
2005-06	CP#28	CP-050063	46		Reference update: xcap-list-usage	6.3.0	6.4.0
2005-06	CP#28	CP-050063	47		Reference update: policy	6.3.0	6.4.0
2005-06	CP#28	CP-050063	48		Reference update: config-framework	6.3.0	6.4.0
2005-06	CP#28	CP-050063	40	1	Editorial corrections	6.3.0	6.4.0
2005-06	CP#28	CP-050063	42	1	xcap-change substitution	6.3.0	6.4.0
2005-09	CP#29	CP-050441	050	3	Data model changes	6.4.0	6.5.0
2005-09	CP#29	CP-050441	051	2	XCAP changes	6.4.0	6.5.0
2005-07	CP-30				Version 7.0.0 created by MCC due to TISPAN references	6.5.0	7.0.0
2006-03	CP-31	CP-060163	053	2	Clarifications to text on presence information	7.0.0	7.1.0
2006-03	CP-31	CP-060109	054	1	Miscellanous Text Changes	7.0.0	7.1.0
2006-03	CP-31	CP-060109	055	1	Reference Change	7.0.0	7.1.0
2006-03	CP-31	CP-060109	056	1	Replace UA-Profile instead of SIP-profile	7.0.0	7.1.0
2006-09	CP-33	CP-060456	063	1	Removal of Editor's notes in 24.141	7.1.0	7.2.0

History

Document history		
V7.0.0	December 2005	Publication
V7.1.0	March 2006	Publication
V7.2.0	September 2006	Publication