H.323 FORUM

Open Community Specification

Presence in H.323 Systems

December 2009

International Multimedia Communications Consortium

Summary

This Recommendation defines the capability and procedures for facilitating the exchange of presence information between devices using pre-existing H.225.0 RAS mechanisms. The Presence feature can be used to subscribe to and be notified of presence changes for the purpose of determining the availability and the methods supported for communication.

Author(s)

Simon Horne, Spranto Australia Pty. Ltd.   s.horne@spranto.com.au

Acknowledgements

Table of Contents

Presence in H.323 Systems

1   Scope

This Recommendation specifies the protocol to exchange presence related information between H.323 devices.

This document proposes a feature for the purpose to allow H.225.0 AliasAddress to subscribe and notify other H.225.0 AliasAddress their willingness and ability to engage in potential communication within a H.323 system.

This document defines the presence feature using the H.323 General Extensibility Framework.

2   References

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

[1]   ITU-T Recommendation H.323 (2009), *Packet-based multimedia communications systems*.

[2]   ITU-T Recommendation H.225.0 (2009), *Call signalling protocols and media stream packetization for packet-based multimedia communication systems*.

[3]   ITU-T Recommendation H.460.1 (2009), *Guidelines for the Use of the Generic Extensible Framework*.

[4]   ITU-T Recommendation H.245 (2009), *Control Protocol for Multimedia Communications*.

[5]   ITU-T Recommendation H.235 (2005), *Security Framework for H series Multimedia Communications*.

3   Terms and Definitions

Presence: Indication of an H.225.0 AliasAddress willingness and ability for potential communication within a H.323 system.

Presentity: H.225.0 AliasAddress as described by presence information

Instruction: Communication between Endpoints and Gatekeepers for the purpose of updating and synchronising presence information.

Subscription: Request or approval to subscribe to receive presence information

Notification: Sending/receiving presence information

4 Abbreviations

For the purpose of this Recommendation the following abbreviations are used.

P	Presentity
PS	Presence Subscription
PN	Presence Notification
PI	Presence Instruction
PD	Presence Identifier
RRQ	Registration Request
RCF	Registration Confirm
SCI	Service Control Indication
SCR	Service Control Response
LRQ	Location Request
LCF	Location Confirm
PLR	Presence Location Request (as defined in Clause 14.2)

5 Feature Description

This Recommendation specifies the methodology to exchange presence related information between H.323 devices.

This document proposes an extension to H.323 for the subscription and notification of alias presence. Alias presence is defined as the willingness and ability of an alias to communicate with other aliases on the network. Historically, presence has been limited to basic "on-line" and "off-line" indicators; the notion of presence here is much broader to include physical presence (location), the modes of communication (audio/video etc) supported and an extensible framework for identifying new and future presence features. This document defines the baseline presence feature using H.323 General Extensibility Framework.

6 Capability Advertisement

H.323 devices capable of supporting Presence shall advertise the support via the Generic Extensibility Framework defined in Recommendations H.323 and H.460.1. Endpoints shall advertise in the GRQ and RRQ message when registering with a gatekeeper as a supported feature. Gatekeepers capable of supporting this feature shall advertise to other gatekeepers when enquiring for this recommendation via the genericData field of the LRQ and the reply in the corresponding LCF messages

Table 1 below defines the Presence feature in this Recommendation.

Table 1 – Indication of the Presence Feature

Feature name:	PresenceFeatureIdentifer
Feature Description:	This feature allows for the collection and notification of status information of elements of a H323 network.
Feature identifier type:	OID
Feature identifier value:	iso(1) org(3) dod(6) internet(1) private(4) enterprise(1) packetizer(17090) gef(0) presence(3)

Parameters associated with the advertisement of this capability are specified in the following clauses. In consideration of backward compatibility with further revisions to this Recommendation, the recipient shall simply ignore any parameters received other than those specified in this document.

7   Overview

Presence shall be defined in terms of H.225.0 AliasAddress supplied or assigned to a H.323 device during registration. The device may contain multiple aliases and each alias may contain different presence information. H.323 device furnishes presence notifications (online,offline,geolocation) and instructions (subscribe,unsubscribe) to the gatekeeper. The gatekeeper then acts on those notifications and instructions. Each subscription has a Globally Unique Identifier created by the gatekeeper. Presence information may be exchanged between gatekeepers using this identifier. Gatekeepers may also use this identifier to determine pathways to forward presence messages to other gatekeepers.

8   Presence Element Message

All messages in this recommendation shall be transmitted via encoded **PresenceElement** contained in a **Presence** Indicator.

Parameter name:	Presence Indicator
Parameter description:	Indicates the current presence state of the entity to the presence service
Parameter identifier type:	Standard
Parameter identifier value:	1
Parameter type:	octetString (encoded PresenceElement)
Parameter status:	Mandatory
Parameter cardinality:	Once and only once

Presence Indicators may be included with the Presence Feature advertisement in the **GRQ**, **RRQ** and **LRQ** messages and the responding **GCF**, **RCF** and **LCF** and included in the **genericData** field of **SCI** messages.

The **PresenceElement** shall contains one or more **PresenceMessage**

9   PresenceMessage

The **PresenceMessage** shall contain the Presence Information for this recommendation. The type of **PresenceMessage** is listed in Table 3.

Table 3 – PresenceMessage types

Message Type	Usage
Status	Subscription information update between endpoint and gatekeeper.
Instruct	Endpoint instructions on presence (subscribe,block)
Authorize	Authorization for presence subscription for Endpoints
Notify	Presence Notification to an Endpoint
Request	Request Presence from another Gatekeeper
Response	Response to Presence for another Gatekeeper
Alive	Keep-Alive message for subscriptions between Gatekeepers
Remove	Notification of removal of Presence subscription from Gatekeeper
Alert	Presence Notification alert from another Gatekeeper

Table 4 indicates the **PresenceMessage** usage in H.225.0 signalling.

Message Type	RRQ	RCF	SCI	LRQ	LCF
Status	X				
Instruct		X	X		
Authorize			X		
Notify			X		
Request				X	
Response				X	
Alive				X	X
Remove				X	
Alert				X	

The **PresenceMessage** structures are defined in cause 12.

10 Presentity Element (P)

The **Presentity** element contains the following:

State: **PresenceState** as per Table 4.

SupportedFeatures: (optional) Sequence of **PresenceFeature** as per Table 5.

GeoLocation: (optional) **PresenceGeoLocation** element as per clause 10.3.

Display: (optional) Sequence of **PresenceDisplay** as per clause 10.4.

GenericData: (optional) Sequence of **GenericData**

10.1 PresenceState

Table 4 – Presence States

PresenceState	Intended Usage
Hidden	Presence appears to other aliases as being offline (not registered) however still receives presence information from other aliases.
Available	Available to receive calls.
Online	Alias has logged in and by default able to receive calls.
Offline	Alias is offline. (not registered)
OnCall	Alias is on a call.
VoiceMail	Alias is sending calls to voice mail. (alias may or may not be registered)

NotAvailable	Alias is registered but not available for calls
Away	Alias is registered however there has been a period of inactivity at the device.
presGeneric	Generic Alias State (**PresenceDisplay** as defined in clause 10.4)

10.2 PresenceFeature

Table 5 – Presence States

PresenceFeature	Intended Usage
Audio	Alias supports Audio
Video	Alias supports Video
Data	Alias supports Data
ExtVideo	Alias supports Application sharing (H.239)
Generic	Alias supports a **PresenceFeatureGeneric** (as defined in clause 10.5)

A **PresenceFeatureGeneric** element shall contain the features H.225.0 **GenericIdentifier** and an **IA5String** representation for display on the remote alias.

10.3 PresenceGeoLocation

The **PresenceGeoLocation** element may contain 1 or more of the following fields (all fields are optional)

- **Locale:** Local area name usually a city

- **Region:** Region name usually a state or province

- **Country:** Country Name

- **CountryCode:** The 3 digit country code (as per ISO 3166-1 alpha-3)

- **Latitude:** Angle on the Earth's surface to the equatorial plane

- **Longitude:** Angle east or west of a reference meridian

- **Elevation:** Height in meters above sea level

10.4 PresenceDisplay

The **PresenceDisplay** element shall contain

- **Language:** A IA5String of the language as defined in RFC4646

- **Display:** Presence message to display

10.5 PresenceFeatureGeneric

The **PresenceFeatureGeneric** element shall contain

- **Identifier:** Generic Identifier of the feature.

- **Display:** String display string of the feature.

11 Primary Presence Elements

A **PresencePDU** is a choice of 4 primary presence elements.

- Presence Instruction (PI)

- Presence Subscription (PS)

- Presence Notification (PN)

- Presence Identifier (PD)

11.1 Presence Instruction (PI)

The Presence Instruction element contains an Alias Address element with the following choices.

subscribe: subscribe to Alias Address.

unsubscribe: unsubscribe to Alias Address.

block: block presence for Alias Address.

unblock: unblock the presence for Alias Address.

pending: pending authorization for Alias Address.

11.2 Presence Subscription (PS)

The Presence Subscription element contains the following:

identifier: PresenceIdentifer of the Subscription

subscribe: Alias Address requesting to be subscribed to.

aliases: Sequence of Alias Address identifying endpoint requesting presence.

approved : (optional) Whether the subscription has been approved

rasAddress: (optional) RAS signaling address to receive reply

timeToLive: (optional) Time to Live (in sec) to confirm active subscriptions

genericData :   (optional) Sequence of Generic Data

11.3 Presence Notification (PN)

The Presence Notification element contains the following:

aliasAddress   : (optional) Alias Address of the endpoint to receive presentity

subscribers   : (optional) Sequence of **PresenceIdentifier** to receive presentity

presentity (P)   : Presentity information

11.4 Presence Identifier (PD)

The Presence Identifier element contains the following:

- **guid** : Globally Unique Identifier.

12 Presence Message Structure

Each **PesenceMessage** type (refer Clause 9) shall contain one or more primary Presence elements (as described in Clause 11) and where required the H.225.0 **AliasAddress** of the intended recipient if sent from the gatekeeper or the sender's H.225.0 **AliasAddress** if sent to the gatekeeper.

Table 5 defines the various compositions of the **PresenceMessage** types.

<p align="center">Table 5 – PresenceMessage Structure</p>

Message Type	AliasAddress	Notification (PN)	Subscription (PS)	Instruction (PI)	Identifier (PD)
Status	S	M		O	
Instruct	M			M	
Authorize	M		M		
Notify	M	M			
Request			M		
Response			M		
Alive					M
Remove					M
Alert		M			

<p align="center">S- Sequence Mandatory M- Mandatory O- Optional</p>

13   Endpoint Presence Procedures

13.1 Endpoint registering for Presence Service

Where an endpoint supporting this feature registers to a gatekeeper, that may also support this feature, the endpoint shall include with the Presence feature advertisement in the RRQ message a **PresenceStatus** message containing one or more **PresenceNotify** element (PN) that includes a **presentity** (P) containing the current **PresenceState** describing the presence of the appropriate H.225.0 AliasAddresses. Optionally the **PresenceStatus** message may include one or more **PresenceInstructions** element (PI) for the purpose of instructing the gatekeeper to the current subscribe or block lists. The gatekeeper may use this information for the purpose of synchronization with presence information stored in the gatekeeper (refer Clause 14.1). The RRQ message may contain one or more **PresenceStatus** element which shall indicate different groups of AliasAddresses are to be handled differently.

Presentity (P) may include one or more **PresenceFeature** indicating the features that the appropriate AliasAddress supports. This is contained in the **supportedFeatures** field.

Physical geographical location information may be included with **Presentity**. This information shall be contained in the **geoLocation** field.

An implementer may wish to include extended **presentity** information by including links to personal schedule, alternate contact information etc which is beyond the scope of this recommendation. This information shall be encoded and contained as an element in the **generic** field.

The gatekeeper shall respond via the RCF that the gatekeeper supports presence and may include an **Instuct** message containing one or more **presenceInstructions** (PI) to update and resynchronise with the endpoint.

13.2 Alias updating local presence

Once an endpoint has successfully registered and the presence has been forwarded to the Gatekeeper in the RRQ, the endpoint may at any time send a fresh presentity element (P) to the gatekeeper to update the presence state. The endpoint shall send this information to the gatekeeper via a SCI message. The updated presence information shall be via a presentity element (P) included in a Presence Notification (PN) contained within a **Notify** message.

13.3 Subscribing to receive other alias presence

An alias may at any time when registered with a gatekeeper, send one or many presence subscribe instructions by sending a SCI message to the gatekeeper containing a Presence Instruction (PI) with the choice set to **subscribe** for each alias address to subscribe to, that is included in a **Instruct** message.

13.4 Unsubscribing from receiving alias presence

An alias may at any time when registered with a gatekeeper unsubscribe from presence for an endpoint by sending a SCI message to the gatekeeper containing one or many Presence Instruction (PI) with the choice set to **unsubscribe** for each alias address to unsubscribe to, that is included in a **Instruct** message.

13.5 Requesting to block presence for an alias

An alias may block sending presence or receiving calls from certain aliases. This request may be sent at any time while registered via an SCI message. To block presence or calls, a message containing one or many Presence Instruction (PI) with the choice set to **block** for each alias address to block. The Presence Instruct (PI) shall be included in an **Instruct** message.

13.6 Requesting to unblock presence for an alias

An alias may at any time when registered with a gatekeeper unblock presence for an alias previously blocked by sending a SCI message to the gatekeeper containing one or many Presence Instruction (PI) with the choice set to **unblock** for each alias address to unblock. The Presence Instruct (PI) shall be included in an **Instruct** message.

13.7 Subscription Authorization

A gatekeeper that receives a presence instruction of an alias shall request from that alias authorization to supply notifications to the requesting party. In this case the gatekeeper shall send to the endpoint with the requested alias, a Presence Subscription (PS) containing a generated globally unique **PresenceIdentifer** to identify the subscription, the alias to subscribe to in the subscribe field and a sequence of aliases requesting subscription. The approve field shall be omitted.

The PS shall be contained in a **PresenceAuthorize** element together with the alias to be subscribed to and be transmitted to the endpoint via an SCI message.

Each Presence Subscription (PS) element shall be stored locally in the endpoint until it has be either authorized or blocked. If the PS has been approved, the endpoint shall resend the received Presence Subscription (PS) including the optional **Approval** field set to TRUE if approved or FALSE if declined. The PS elements shall be contained in the **subscription** field of a **PresenceAuthorize** element together with the first alias requesting the subscription and be transmitted to the gatekeeper via an SCI message.

13.8 Endpoint Presence Notification

An endpoint may receive presence updates for an alias via an SCI message from the gatekeeper containing a **PresenceNotify** element containing one or more **PresenceNotification** (PN) as well as the alias to receive the presence notification.

14 Gatekeeper Presence Procedures

14.1 Storage of Subscription and Block list Information

A gatekeeper may select to store endpoint subscription and block list information on behalf of the endpoint. In this case, when an endpoint first registers with the gatekeeper, the gatekeeper shall check the supplied subscriptions and block lists received from the endpoint with the corresponding information stored on in memory or database. The gatekeeper shall send to the endpoint the full list of active or pending (outbound) subscriptions via the replying RCF message. The message shall be in form of one or more **PresenceInstruction** (PI) contained with in a **PresenceInstruct** element for each applicable registering alias.

Subscriptions requiring approval shall be transmitted to the registering endpoint via one or more **PresenceSubscription** (PS) contained with in a **PresenceAuthorize** element for each applicable registering alias.

How gatekeepers stores subscription and block information is outside the scope of this recommendation.

14.2 Subscriptions for aliases on other gatekeepers

When subscribing to aliases on endpoints registered on other gatekeepers, the LRQ message shall be used to handle all presence exchanges.

Presence subscriptions between gatekeepers shall involve 6 steps

- Alias discovery
- Request for subscription
- Response to request
- Alerting of presence changes
- Keeping subscriptions active
- Removal of presence subscription

To provide scalability after the alias discovery process, the LRQ may contain multiple presence elements included as a single **PresenceElement** contained in an empty LRQ (no **endpointIdentifier** and no **destinationInfo**). This shall be referred to as a Presence location Request or **PLR** for the purpose of this clause.

14.2.1 Alias discovery

Where an alias cannot be resolved as being registered to the local gatekeeper, the gatekeeper may use the pre-existing LRQ mechanism to resolve the address of an alias requested to be subscribed to. The LRQ message shall, within the **genericData** field of the PDU, include the **PresenceFeature Identifer** containing the **PresenceDiscoveryIdentifer**. The **PresenceDiscoveryIdentifer** shall contain an encoded **PresentElement** that includes one or more **PresenceAlive** elements. The **PresenceAlive** element shall contain the **PresenceIdentifier** of the existing or proposed subscription associated with the alias contained within the **sourceInfo** element of the LRQ.

Table 6 –Presence Discovery Parameter

Parameter name:	Presence Discovery Identifier
Parameter description:	Discovery element to resolve location of alias subscriptions
Parameter identifier type:	Standard
Parameter identifier value:	2
Parameter type:	octetString (encoded PresenceElement)
Parameter status:	Mandatory
Parameter cardinality:	Once and only once

Table 7 –Presence TTL Parameter

Parameter name:	PresenceTTLIdentifier
Parameter description:	Time To Live (sec) for subscriptions for the subscribed alias
Parameter identifier type:	Standard
Parameter identifier value:	3
Parameter type:	Integer32
Parameter status:	Mandatory
Parameter cardinality:	Once and only once

The requested alias may or may not be currently registered with the gatekeeper. Where a gatekeeper does not currently have the alias registered and the gatekeeper knows is responsible for the presence of that alias, the gatekeeper may act on the alias's behalf and respond back with an LCF.

The responding LCF shall contain within the **genericData** field of the PDU, the **PresenceFeature Identifer** containing the **PresenceDiscoveryIdentifier** and **PresenceTTLIdentifier**. The **PresenceDiscoveryIdentifier** shall contain an encoded **PresentElement** that includes one or more **PresenceAlive** elements. The **PresenceAlive** element shall contain the **PresenceIdentifier** of the existing or proposed subscription associated with the alias contained within the **destinationInfo** field of the LCF. The **rasAddress** element of the LCF shall contain the RAS address of the gatekeeper that is to handle presence messages. The **PresenceTTLIdentifier** shall contain the length of time the subscription will be active without being renewed.

On receipt of the LCF the gatekeeper shall record that the alias can be located at the **rasAddress** element of the LCF. Where the LCF received is not the final destination, ie not the final hop, the gatekeeper shall record that a **PresenceIdentifier** received shall be passed onto the address specified in the **rasAddress** field of the LCF or forwarded to the RAS Address of the forwarded LCF. The record shall be active for the length specified in the PresenceTTLIdentifer. In this way presence may be supported and maintained within a hierarchical gatekeeper structure.

14.2.2 Request for subscription
Where an alias has been discovered as being registered to another gatekeeper, the gatekeeper shall send any initiating **PresenceSubscription** (PS) via a **PresenceRequest** element contained in a PLR. The receiving gatekeeper shall forward this element to the endpoint registered with the alias contained in the **subscribe** field of the PS via a **PresenceAuthorize** element contained in an SCI. Where that endpoint is not registered it shall store the PS until the endpoint registers and then forward the PS to the endpoint.

14.2.3 Response to subscription
Where an endpoint has sent a **PresenceAuthorize** element containing a **PresenceSubscription** (PS) to the gatekeeper that includes the optional field **Approved**, the gatekeeper shall resolve the destination to forward the PS onto. If the alias is not registered (or managed) locally, the gatekeeper shall initiate the Alias discovery process as described in clause 14.2.1. Once a RAS address has been resolved the PS is to be sent via a **PresenceResponse** element contained in a PLR to the resolved RAS address. The PS shall include the **timeToLive** field indicating the length of time (in seconds) the subscription will be held active before requiring renewal.

14.2.4 Alerting of presence changes
Where an endpoint registers with an existing subscription managed by another gatekeeper, the gatekeeper shall initiate the Alias discovery process as described in clause 14.2.1. Once a RAS address has been resolved the full **presentity** (P) of the registering alias shall be forwarded to the resolved RAS address via a **PresenceNotification** (PN) contained in a **PresenceAlert** element of a PLR. Any subsequent presence changes for the alias from the endpoint shall be sent via the same procedure.

14.2.5 Keeping subscriptions active

Subscriptions are active only for the time specified in the **PresenceTTLIdentifier** sent in the responding LCF during alias discovery. The subscribing gatekeeper must send a **PresenceAlive** element containing the **PresenceIdentifier** (PI) via a PLR before the **PresenceTTLIdentifier** expires. Once a TTL has expired the subscription shall be considered dormant until a fresh Alias discovery is made. Numerous **PresenceAlive** elements may be aggregated into 1 PLR for scalability. Upon receipt any intermediary gatekeepers shall also renew the subscription pathway before passing the PLR on.

14.2.6 Removal of presence subscription

When either endpoint requests the removal of a subscription of an alias then it shall follow the procedures specified in clause 13.4. If the gatekeeper determines that the subscription is for an alias registered to another gatekeeper then the gatekeeper shall send an **PresenceRemove** element containing the **PresenceIdentifier** (PI) of the subscription to remove and optionally the **remove** field set to TRUE.

15 Presence Message Aggregation

All presence messages are designed to aggregate for greater scalability. Presence messages may be aggregated based on the endpoint or transport RAS address they are to be set to. The implementer may choose to limit the number of messages sent across the network and only send messages every few seconds. The degree of presence message aggregation is beyond the scope of this document and shall be left to the discretion of the implementer.

16 Security considerations

It is recommended that an implementer employ the pre-existing H.235 security mechanisms available within RAS messaging to secure presence messages.

Appendix A: ASN.1

```
H460PRES {iso(1) org(3) dod(6) internet(1) private(4) enterprise(1) packetizer(17090) gef(0)
presence(3)} DEFINITIONS AUTOMATIC TAGS ::=

BEGIN

IMPORTS
            AliasAddress,
            TransportAddress,
            TimeToLive,
            GenericData,
            GenericIdentifier,
            GloballyUniqueID
       FROM MULTIMEDIA-SYSTEM-CONTROL;

--
-- Presence Base Message
--

PresenceElement ::= SEQUENCE
{
      message         SEQUENCE OF PresenceMessage,
      ...
}


--
-- Presence Message
--

PresenceMessage ::= CHOICE
{
      presenceStatus        PresenceStatus,
      presenceInstruct      PresenceInstruct,
      presenceAuthorize     PresenceAuthorize,
      presenceNotify        PresenceNotify,
      presenceRequest            PresenceRequest,
      presenceResponse      PresenceResponse,
      presenceAlive         PresenceAlive,
      presenceRemove        PresenceRemove,
      presenceAlert         PresenceAlert,
      ...
}


PresenceStatus ::= SEQUENCE
{
      alias               SEQUENCE OF AliasAddress,
      notification   SEQUENCE OF PresenceNotification,
      instruction         SEQUENCE OF PresenceInstruction          OPTIONAL,
      ...
}

PresenceInstruct ::= SEQUENCE
{
      alias               AliasAddress,
      instruction         SEQUENCE OF PresenceInstruction,
      ...
}

PresenceAuthorize ::= SEQUENCE
{
      alias               AliasAddress,
      subscription        SEQUENCE OF PresenceSubscription,
      ...
}

PresenceNotify ::= SEQUENCE
{
      alias               AliasAddress,
      notification        SEQUENCE OF PresenceNotification,
```

```
        ...
}

PresenceRequest ::= SEQUENCE
{
        subscription            SEQUENCE OF PresenceSubscription,
        ...
}

PresenceResponse ::= SEQUENCE
{
        subscription            SEQUENCE OF PresenceSubscription,
        ...
}

PresenceAlive ::= SEQUENCE
{
        identifier              SEQUENCE OF PresenceIdentifier,
        ...
}
PresenceRemove ::= SEQUENCE
{
        identifier              SEQUENCE OF PresenceIdentifier,
        ...
}

PresenceAlert ::= SEQUENCE
{
        notification            SEQUENCE OF PresenceNotification,
        ...

}


--
-- Presence PDU
--

PresencePDU ::= CHOICE
{
        instruction             PresenceInstruction,
        notification            PresenceNotification,
        subscription            PresenceSubscription,
        identifier              PresenceIdentifier,
        ...
}


--
-- Presence Instruction
--

PresenceInstruction ::= CHOICE
{
        subscribe               AliasAddress,
        unsubscribe             AliasAddress,
        block                   AliasAddress,
        unblock                 AliasAddress,
        pending                 AliasAddress,
        ...
}


--
-- Presence Subscription
--

PresenceSubscription ::= SEQUENCE
{
        identifier              PresenceIdentifier,
        subscribe               AliasAddress,
        aliases                 SEQUENCE OF AliasAddress,
        approved                BOOLEAN                         OPTIONAL,
        rasAddress              TransportAddress                OPTIONAL,
        timeToLive              TimeToLive                      OPTIONAL,
        genericData             SEQUENCE OF GenericData                 OPTIONAL,
        ...
```

```
}


--
-- Presence Identifier
--

PresenceIdentifier ::= SEQUENCE
{
        guid                    GloballyUniqueID,
        remove                  BOOLEAN                         OPTIONAL,
        ...
}


--
-- Presence Notification
--

PresenceNotification ::= SEQUENCE
{
        aliasAddress    AliasAddress                            OPTIONAL,
        subscribers             SEQUENCE OF PresenceIdentifier          OPTIONAL,
        presentity              Presentity,
        ...
}


--
-- Presentity
--

PresenceDisplay ::= SEQUENCE
{
        language                IA5String                       OPTIONAL,       -- RFC4646 language
tag
        display                 BMPString(SIZE(1..128))
}


PresenceState ::= CHOICE
{
        hidden          NULL,
        available       NULL,
        online          NULL,
        offline         NULL,
        onCall          NULL,
        voiceMail       NULL,
        notAvailable    NULL,
        away                    NULL,
        ...,
        generic         SEQUENCE OF PresenceDisplay

}

PresenceFeatureGeneric ::= SEQUENCE
{
        identifier      GenericIdentifier,
        display         IA5String               OPTIONAL,
        ...
}

PresenceFeature ::= CHOICE

{
        audio           NULL,
        video           NULL,
        data            NULL,
        extVideo        NULL,
        ...,
        generic         PresenceFeatureGeneric

}


PresenceGeoLocation ::= SEQUENCE
{
```

```
        locale          IA5String       OPTIONAL,
        region          IA5String       OPTIONAL,
        country         IA5String       OPTIONAL,
        countryCode     IA5String       OPTIONAL,
        latitude        IA5String       OPTIONAL,
        longitude       IA5String       OPTIONAL,
        elevation       IA5String       OPTIONAL,
        ...
}


Presentity ::= SEQUENCE
{
        state                   PresenceState,                                  -- Presence State
        supportedFeatures       SEQUENCE OF PresenceFeature     OPTIONAL,       -- Features
Supported by Endpoint
        geolocation             PresenceGeoLocation            OPTIONAL,        -- Geographic
Location
        display                 SEQUENCE OF PresenceDisplay     OPTIONAL,       -- Basic ISO/IEC
10646-1 (Unicode)
        genericData             SEQUENCE OF GenericData         OPTIONAL,       -- Generic Presence
Information
        ...
}

END
```

Appendix B: Sample Message Flows

This Appendix describes a simple presence exchange between 2 endpoints registered on the same gatekeeper.

```
EP1 Registers
RRQ
This message indicates to the endpoint that 2 aliases have the same presence notification as
well as instructions to subscribe and block aliases.

message = 1 entries {
      [0]=presenceStatus {
        alias = 2 entries {
          [0]=dialedDigits "61001"
          [1]=url_ID "s.horne@packetizer.com"
        }
        notification = 1 entries {
          [0]={
            presentity = {
              state = online <<null>>
              supportedFeatures = 2 entries {
                [0]=audio <<null>>
                [1]=video <<null>>
              }
              geolocation = {
                locale = "Brisbane"
                region = "Queensland"
                country = "Australia"
                latitude = "27.24"
                longitude = "153.02"
              }
              display = 1 entries {
                [0]={
                  language = "en-US"
                  display =  31 characters {
                    0049 0027 006d 0020 006f 006e 006c 0069   I'm onli
                    006e 0065 0020 0061 006e 0064 0020 0072   ne and r
                    0065 0061 0064 0079 0020 0066 006f 0072   eady for
                    0020 0063 0061 006c 006c 0021 0000         call!
                  }
                }
              }
            }
          }
        }
        instruction = 2 entries {
          [0]=subscribe dialedDigits "61002"
          [1]=block dialedDigits "61005"
        }
      }
    }


RCF
Contains instructs telling the endpoint the status of all current instructions. Note 61001 has
a pending subscription to 61002

    message = 2 entries {
      [0]=presenceInstruct {
        alias = dialedDigits "61001"
        instruction = 2 entries {
          [0]=pending dialedDigits "61002"
          [1]=block dialedDigits "61005"
        }
      }
      [1]=presenceInstruct {
        alias = url_ID "s.horne@packetizer.com"
        instruction = 2 entries {
          [0]=subscribe dialedDigits "61002"
          [1]=block dialedDigits "61005"
        }
      }
```

```
        }

EP2 Registers
RRQ
Indicates that the endpoint has 1 alias and has 1 blocking instruction

      message = 1 entries {
        [0]=presenceStatus {
          alias = 1 entries {
            [0]=dialedDigits "61002"
          }
          notification = 1 entries {
            [0]={
              presentity = {
                state = online <<null>>
                supportedFeatures = 2 entries {
                  [0]=audio <<null>>
                  [1]=video <<null>>
                }
                geolocation = {
                  locale = "Brisbane"
                  region = "Queensland"
                  country = "Australia"
                  latitude = "27.24"
                  longitude = "153.02"
                }
                display = 1 entries {
                  [0]={
                    language = "en-US"
                    display =  31 characters {
                      0049 0027 006d 0020 006f 006e 006c 0069   I'm onli
                      006e 0065 0020 0061 006e 0064 0020 0072   ne and r
                      0065 0061 0064 0079 0020 0066 006f 0072   eady for
                      0020 0063 0061 006c 006c 0021 0000         call!
                    }
                  }
                }
              }
            }
          }
          instruction = 2 entries {
            [0]=block dialedDigits "61005"
          }
        }
      }

RCF
Indicates to EP2 that it has a block instruction as well as an existing subscription stored in
the database (bold).

      message = 1 entries {
        [0]=presenceInstruct {
          alias = dialedDigits "61002"
          instruction = 2 entries {
            [0]=block dialedDigits "61005"
            [1]=subscribe url_ID "s.horne@packetizer.com"
          }
        }
      }

SCI from GK to EP2
This message contains 2 elements. An authorization element to subscribe to 61001 and a
notification element containing the full presence information for subscription to
s.horne@packetizer.com

      message = 2 entries {
        [0]=presenceAuthorize {
          alias = dialedDigits "61002"
          subscription = 1 entries {
            [0]={
              identifier = {
                guid =  16 octets {
                  ec 3e 68 d7 26 00 19 10  8b 91 00 01 02 92 d2 3b   .>h.&..........;
                }
              }
              subscribe = dialedDigits "61002"
```

```
            aliases = 1 entries {
               [0]=dialedDigits "61001"
            }
         }
       }
     }
   [1]=presenceNotify {
     alias = dialedDigits "61002"
     notification = 1 entries {
        [0]={
          aliasAddress = url_ID "s.horne@packetizer.com"
          presentity = {
            state = online <<null>>
            supportedFeatures = 2 entries {
               [0]=audio <<null>>
               [1]=video <<null>>
            }
            geolocation = {
               locale = "Brisbane"
               region = "Queensland"
               country = "Australia"
               latitude = "27.24"
               longitude = "153.02"
            }
            display = 1 entries {
               [0]={
                 language = "en-US"
                 display =  31 characters {
                    0049 0027 006d 0020 006f 006e 006c 0069   I'm onli
                    006e 0065 0020 0061 006e 0064 0020 0072   ne and r
                    0065 0061 0064 0079 0020 0066 006f 0072   eady for
                    0020 0063 0061 006c 006c 0021 0000          call!
                 }
               }
            }
          }
        }
     }
   }
 }
```

SCI from EP2 to GK
EP2 sends an authorization element with the optional field approved set to true. (false means subscription denied).

```
   message = 1 entries {
     [0]=presenceAuthorize {
       alias = dialedDigits "61002"
       subscription = 1 entries {
          [0]={
            identifier = {
               guid =  16 octets {
                 ec 3e 68 d7 26 00 19 10  8b 91 00 01 02 92 d2 3b   .>h.&..........;
               }
            }
            subscribe = dialedDigits "61002"
            aliases = 1 entries {
               [0]=dialedDigits "61001"
            }
            approved = true
          }
       }
     }
   }
```

```
SCI from GK to EP1
Message contains 2 elements. An instruction notifying that a subscription to 61002 now exists
for 61001 and the full presence information for that subscription.

    message = 2 entries {
      [0]=presenceInstruct {
        alias = dialedDigits "61001"
        instruction = 1 entries {
          [0]=subscribe dialedDigits "61002"
        }
      }
      [1]=presenceNotify {
        alias = dialedDigits "61001"
        notification = 1 entries {
          [0]={
            aliasAddress = dialedDigits "61002"
            presentity = {
              state = online <<null>>
              supportedFeatures = 2 entries {
                [0]=audio <<null>>
                [1]=video <<null>>
              }
              geolocation = {
                locale = "Brisbane"
                region = "Queensland"
                country = "Australia"
                latitude = "27.24"
                longitude = "153.02"
              }
              display = 1 entries {
                [0]={
                  language = "en-US"
                  display =  31 characters {
                    0049 0027 006d 0020 006f 006e 006c 0069   I'm onli
                    006e 0065 0020 0061 006e 0064 0020 0072   ne and r
                    0065 0061 0064 0079 0020 0066 006f 0072   eady for
                    0020 0063 0061 006c 006c 0021 0000         call!
                  }
                }
              }
            }
          }
        }
      }
    }


SCI from GK to EP2
Message contains 2 elements. An instruction notifying that a subscription to 61001 now exists
for 61002 and the full presence information for that subscription.

    message = 2 entries {
      [0]=presenceInstruct {
        alias = dialedDigits "61002"
        instruction = 1 entries {
          [0]=subscribe dialedDigits "61001"
        }
      }
      [1]=presenceNotify {
        alias = dialedDigits "61002"
        notification = 1 entries {
          [0]={
            aliasAddress = dialedDigits "61001"
            presentity = {
              state = online <<null>>
              supportedFeatures = 2 entries {
                [0]=audio <<null>>
                [1]=video <<null>>
              }
              geolocation = {
                locale = "Brisbane"
                region = "Queensland"
                country = "Australia"
                latitude = "27.24"
                longitude = "153.02"
              }
              display = 1 entries {
```

```
                    [0]={
                      language = "en-US"
                      display =  31 characters {
                        0049 0027 006d 0020 006f 006e 006c 0069   I'm onli
                        006e 0065 0020 0061 006e 0064 0020 0072   ne and r
                        0065 0061 0064 0079 0020 0066 006f 0072   eady for
                        0020 0063 0061 006c 006c 0021 0000           call!
                      }
                    }
                  }
                }
              }
            }
          }
        }
```

SCI from GK to EP2
Lightweight Notification to 61001 and <u>s.horne@packetizer.com</u> that 61002 has gone offline

```
      message = 2 entries {
        [0]=presenceNotify {
          alias = dialedDigits "61001"
          notification = 1 entries {
            [0]={
              aliasAddress = dialedDigits "61002"
              presentity = {
                state = offline <<null>>
              }
            }
          }
        }
        [1]=presenceNotify {
          alias = url_ID "s.horne@packetizer.com"
          notification = 1 entries {
            [0]={
              aliasAddress = dialedDigits "61002"
              presentity = {
                state = offline <<null>>
              }
            }
          }
        }
      }
```