SIMPLE WG
Internet-Draft
Expires: July 26, 2004

T. Moran

H. Khartabil
E. Leppanen
Nokia

January 26, 2004

Requirements for Presence Specific Event Notification Filtering
draft-ietf-simple-pres-filter-reqs-03

Status of this Memo

Copyright Notice

Abstract

This document defines a set of structured requirements whereby a presence information subscriber may select specific information to be received in the presence information notification sent by the notifier. The purpose is to limit the content and frequency of notifications so that only essential information on a need basis is delivered by the server.

Table of Contents

1. Introduction

SIP event notification is described in [6]. It defines a general framework for subscriptions and notifications for SIP event packages. Concrete applications of the general event framework to a specific group of events are described in [5] (user presence) and [7] (watcher information).

The presence information refers to a set of presence attributes describing the availability and willingness of the user (presentity) for communication. The user makes his presence information available for other users (watchers).

As the inherent usage of event packages grows, the client needs some mechanisms for controlling the event notifications at the source. Evidence of this need is found in [4].

The document describing the Presence event package [5] mentions the possibility for filtering. Accordingly, the SUBSCRIBE request may contain a body for filtering the presence information subscription. However, the definition of filtering was considered out of scope and was left as future work.

These mechanisms are expected to be particularly valuable to users of wireless devices. The characteristics of these devices typically include low bandwidth, low data processing capabilities, small display and limited battery power. Such devices can benefit from the ability to filter the amount of information generated at the source of the event notifications.

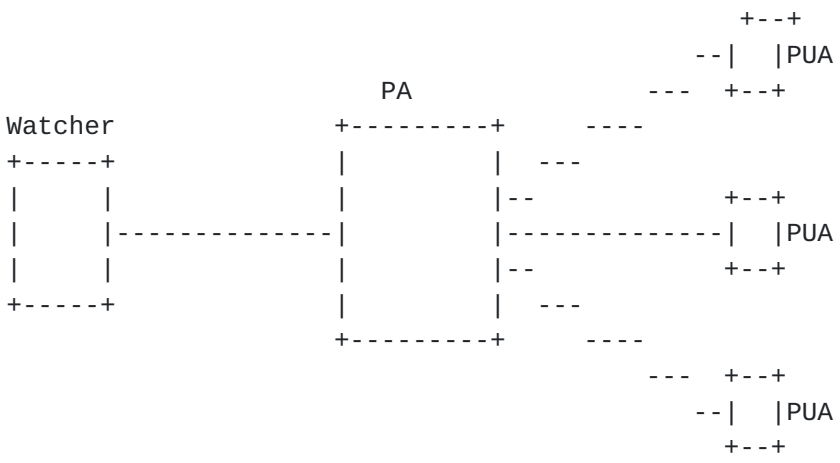
However, it is expected that the control mechanisms for event notifications add value for all users irrespectively of their device or network access characteristics.

Section 4 and Section 5 of this draft propose a set of requirements whereby a client may specify which notifications it is interested in. That is, a means to specify filtering rules to be executed by the server. Section 8 provides a few example applications of notification filtering.

2. Event Filtering Model

There are two parts to the event filtering model. From a Presence service view point, presence information is collected by a Presence Agent and is published by one or more Presence User Agents. The first part of the model enables the watcher to limit the presence information delivered to it. Allowing the watcher to select the information of interest to it results in the ability to limit the

contents of a presence information document, therefore reducing the size of a notification message.



The second part of the model defines the triggering. In a filter-less subscription, it might be a Presence Agent's default policy to deliver a notification message every time there is a change to the presence information of a presentity or whenever a PUA publishes new and updated presence information from its own point of view. This model enables the watcher to select the events or changes in presents information that trigger notifications to be sent. Other changes that are not defined as triggers in a filter do not result in a notification message being delivered to the watcher.

3. Conventions

In this document, the key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL', 'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED', 'MAY', and 'OPTIONAL' are to be interpreted as described in RFC 2119 [1] and indicate requirement levels for compliant implementations.

4. Requirements for Specification of Filters

The following requirements relate to the creation of filters.

4.1 Package Identification

REQ A1: It MUST be possible for the creator of the filter to specify the package the filter applies to.

4.2 Target URI

REQ A2: It MUST be possible for the watcher to indicate, in the

filter, the target presentity whose presence information a certain filter is applied to.

REQ A3: It MUST be possible for the watcher to indicate, in the filter, the target domain that a certain filter is applied to. For example an event list might have many resources from different domains, a watcher needs to be able to set a filter for one of those domains.

4.3 Notification Triggering

This chapter presents requirements for specifying the triggering conditions that result in notifications to be sent to the watcher.

REQ B1: The triggering conditions MUST be based on the presence information. For example, the change of value of the <status> element.

REQ B2:It MUST be possible to define a set of conditions for the values of certain elements in a presence document that determine when to send notifications.

REQ B3: It MUST be possible to construct one filter that combines multiple triggering conditions.

4.4 Notification Content

This chapter presents requirements for specifying the filter for choosing content to be sent in the notifications.

REQ C1: It MUST NOT be possible to break any server side policy constraints when applying the content filter. For example, it must not be possible for a watcher to request a notification to contain the <contact> element of a certain presentity when there is a local server policy constraining the delivery of the <contact> element.

REQ C2: It MUST be possible for the watcher to specify the presence information elements (XML elements and/or attributes) in [2] to be delivered in the notification.

REQ C3: It MUST be possible for the watcher to specify presence information in any extension to PIDF to be delivered in the notifications, based on XML elements and/or attributes. See for example [3].

REQ C4: It MUST be possible for the watcher to specify presence information in any extension to be delivered in the notifications, based on namespaces.

REQ C5: It MUST be possible to construct one filter that combine multiple elements and attributes to be included the notifications.

REQ C6: It MUST be possible for the watcher to specify presence information in PIDF or any any extension to it to be excluded from the notifications, based on elements and/or attributes.

5. Requirements for Uploading Filters (Operational Rules)

REQ D1: It MUST be possible for the watcher to upload filters to the server (notifier) and know the status - accepted or rejected, if the server policy allows.

5.1 Subscription

REQ D2: It MUST be possible to place a filter in the body of the SUBSCRIBE request.

REQ D3: It MAY be possible to deliver a filter to a server using other means. For example, it may be possible for the filter to be (permanently) stored in the server.

5.1.1 Maintaining a Filter

REQ D4: The watcher MUST NOT be required to re-set a filter at any time during the subscription, once the filter has been set. This includes subscription refreshes

REQ D5: modifying a filter across subscription refreshes SHOULD be bandwidth efficient.

REQ D6: It MUST NOT be required for a watcher to explicitly remove a filter if the subscription was terminated or has expired. I.e. The filter is automatically removed with the subscription.

5.1.2 Changing a Filter

REQ E1: It MUST be possible to change the filter during a subscription.

REQ E2: It MUST be possible for the watcher to remove a set filter, reverting back to a server defined default.

5.2 Server Support For Filters

REQ F1: It MUST be possible for a server not supporting filtering to inform the watcher of the failure.

REQ F2: It MUST be possible for a server not understanding a filtering to inform the watcher of the failure.

REQ F3: It MUST be possible for a server not accepting a filter to inform the watcher of the reasons for not accepting the filter.

REQ F4: It MUST be possible for the server to terminate a subscription if a filter is no longer acceptable, e.g., due to policy change or server load.

6. Interaction with Other Features

6.1 Resource Lists

REQ G1: It MUST be possible to support filtering for subscriptions to event lists [8].

REQ G2: It MUST be possible for the watcher to indicate, in the filter, the target event list that a certain filter is applied to by the Resource List Server.

REQ G3: It MUST be possible for a watcher to specify individual filters for any resource in an event list if the subscription is for an event list.

REQ G4: It MUST be possible to specify a filter for an event list and a filters for resources within that list in the same subscription request.

REQ G5: Some event lists may contain an other event list as a resource. I.e. nested lists. It MUST be possible for the watcher to indicate, in the filter, a nested event list that a certain filter is applied to.

REQ G6: It MUST be possible for a watcher to specify different filter for resources within any nested list of an event list, if the subscription is for an event list.

REQ G7: It MUST be possible for each watcher to define his/her own filters within an event list subscription if there are several simultaneous watchers using the same list.

6.2 Partial Notifications

REQ H1: It MUST be possible to use filtering along with the partial notification [9] within the same subscription.

6.3 Authorization

7. Security Considerations

Security requirements specified for [5] also applies to presence

filtering. Additional security considerations are described as follows.

REQ I1: It SHOULD be possible for the server to hide the fact that a filter was not acceptable.

REQ I2: The presence of filters in the body in a SIP message has a significant effect on the way in which the request is handled at a server. As a result, it MUST be possible to authenticate messages carrying filters and authorise the watcher to upload filters.

REQ I3: Modification to filters by an intermediary could also result in the watcher either not receiving notifications of presence information they are interested in or receiving a very large presence document. Therefore the filters SHOULD be integrity protected between those nodes that are authorised to modify it (e.g., the resource list servers).

REQ I4: Processing of requests and looking up filters requires some amount of computation. This enables a DoS attack whereby a user can send requests with substantial numbers messages with large contents, in the hopes of overloading the server. To prevent this the number of filters allowed in a request should be limited.

REQ I5: Requests containing filters can reveal sensitive information about a UA's capabilities. If this information is sensitive, it SHOULD be encrypted using methods that allow it to be read by those nodes that need to do so (e.g., the resource list servers).

REQ I6: The resource list servers SHOULD NOT forward filters targeted to a different domain than a fanned out subscription request.

REQ I7: Authorization SHOULD occur irrespective of the filtering.

8. Example Applications for Notification Filtering

- A watcher wishes to get to know presentity's availability and willingness for messaging (e.g. IM and MMS).

- A watcher is interested in getting information about the communication means and contact addresses the presentity is currently available for communication.

- A watcher requires a notification if the state of a buddy has changed to 'open'.

- A watcher only wants to be notified when the presentity's location is Dallas or Fort Worth. The notification should include the

vehicle license, driver name, and city.

- A Basic location tracking service requires notification when the presentity's cell id changes. The notification should include the cell id.

- A watcher is interested in being notified when a presentity gains a new communication capability such as a new networked multi-player game.

9. Acknowledgements

The authors would like to thank Andrew Allen, Sreenivas Addagatla, Mikko Lonnfors, Juha Kalliokulju, Aki Niemi, Jose Costa-Requena, Markus Isomaki, Paul Kyzivat, Aki Niemi and Chris Boulton for their valuable input.

10. Changes from previous versions

10.1 Main changes from version 02

- Added filtering model section.

- Rephrased some requirements for clarity.

- Rearranged requirements into more appropriate sections.

10.2 Main changes from version 01

- "Overview of Operation" section removed .

- "Common Syntax" section removed.

- "Discovery of Items" section removed as agreed in IETF 57

- Added requirement about filtering using namespaces.

- Added requirement about filtering using domain name.

- Clarified and split larger requirements into smaller more concrete requirements.

- Updated the Authors of this ID

10.3 Main changes from version 00

- Overview of functionality chapter added.
- More specific requirements for supporting filtering with the resource lists, and nested lists.
- Interaction with other features chapter added.
- More specific requirements to support getting information about the structure of presence document, and changes in it.
- Several filter specific additions to security considerations.
- Several editorial changes, e.g., reference and terminology updates.

References

[1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[2] Sugano, H., "CPIM Presence Information Data Format", draft-ietf-impp-cpim-pidf-08.txt, May 2003.

[3] Schulzrinne, H., "RPID -- Rich Presence Information Data Format", draft-ietf-simple-rpid-00.txt, July 2003.

[4] Kiss, K., "Requirements for Presence Service based on 3GPP specifications and wireless environment characteristics", draft-kiss-simple-presence-wireless-reqs-02, February 2003.

[5] Rosenberg, J., "Session Initiation Protocol (SIP) Extensions for Presence", draft-ietf-simple-presence-10.txt, January 2003.

[6] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[7] Rosenberg, J., "A Watcher Information Event Template-Package for the Session Initiation Protocol (SIP)", draft-ietf-simple-winfo-package-05.txt, January 2003.

[8] Roach, A., "A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists", draft-ietf-simple-event-list-03.txt, June 2003.

[9] Lonnfors, M., "Partial Notification of Presence Information", draft-lonnfors-simple-partial-notify-01.txt, May 2003.

Authors' Addresses

Tim Moran
2800 Britt Drive
Argyle, Texas 76226
USA

Phone: +1 972 849 8821
EMail: tl_moran@att.net

Hisham Khartabil
Nokia
P.O BOX 321
Helsinki
Finland

Phone: +358 7180 76161
EMail: hisham.khartabil@nokia.com

Eva Leppanen
Nokia
P.O BOX 785
Tampere
Finland

Phone: +358 7180 77066
EMail: eva-maria.leppanen@nokia.com

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.