SIMPLE WG
Internet-Draft
Expires: July 26, 2004

K. Kiss
E. Leppanen
H. Khartabil
Nokia

January 26, 2004

Requirements for Filtering of Watcher Information
draft-ietf-simple-winfo-filter-reqs-01

Abstract

This document defines a set of structured requirements whereby a watcher information subscriber (client) may select specific information to be received in the watcher information notification sent by the notifier (server). The purpose is to limit the content so that only essential information is delivered by the server.

Table of Contents

1. Introduction

SIP event notification is described in [4]. 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 [3] (user presence) and [5] (watcher information).

The watcher information refers to the set of users subscribed to a particular resource within a particular event package. Watcher information changes dynamically as users subscribe, unsubscribe, are approved, or rejected. A client can subscribe to this information.

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 [2].

The Internet Draft describing the watcher information template package [5] mentions the possibility for filtering. Accordingly, the SUBSCRIBE request may contain a body for filtering the watcher information subscription. However, the definition of filtering has been left out of the scope of the Internet Draft.

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 notification.

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

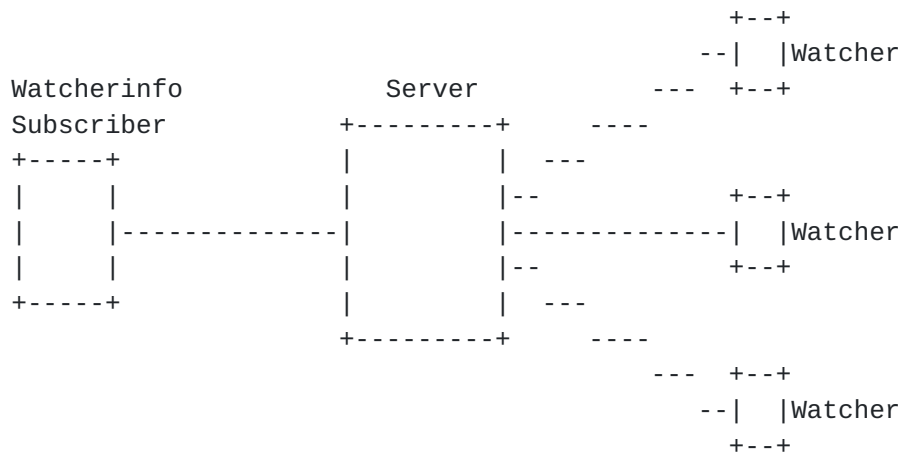
Section 4 and Section 6 of this draft propose a set of requirements whereby a client may specify which notifications it is interested in. That is, a means to specify filtering rules to be executed by the server. Section 8 provides a few example applications of notification filtering.

2. 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.

3. Event Filtering Model

There are two parts to the event filtering model. From a Watcher Information service view point, Watchers subscribe for information about a resource. That resource, called watcherinfo subscriber, sometimes needs to collect information about the watchers. The first part of the model enables the watcherinfo subscriber to limit the watcher information delivered to it. Allowing the watcherinfo subscriber to select the information of interest to it results in the ability to limit the contents of a watcher 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 server's default policy for the first NOTIFY after a subscription to carry full state, and subsequent notifications to carry partial state. This model enables the watcherinfo subscriber to select the events or changes in watcher information that trigger notifications to be sent. Other changes in watcher state that are not defined as triggers in a filter do not result in a notification message being delivered to the watcher.

4. Requirements for Specification of Filters

The following requirements relate to the creation of filters (rules).

REQ A1: A common set of constructs MUST be defined for the creation of rules. There MUST be a common set of operations that follow a common syntax. It MUST be possible for the client to define different rules for different purposes using a common filtering mechanism.

REQ A2: It MUST be possible for the client to indicate the event package to which the filter applies.

REQ A3: It MUST be possible for the client to indicate the target resource to which a certain filter is applied.

5. Notification Content Limiting

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

REQ B1: It MUST be possible for the client to specify the XML elements in winfo format [6] to be delivered in the notification.

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

REQ B3: It MUST be possible to construct expressions that combine multiple tests.

REQ B4: The client MUST be able to instruct the server to include, in the notifications, only those watchers that are in a specific state.

REQ B5: The client MUST be able to indicate to the server to include only those watchers in the notifications which are transitioned in their current status because of a specific event.

REQ B6: The client MUST be able to indicate to the server to include only those watchers in the notifications which have subscription lifetime higher than (less than) a specific amount of seconds.

REQ B7: The client MUST be able to indicate to the server to include only those watchers in the notifications which are subscribed for a duration higher than (less than) a specific amount of seconds.

REQ B8: The client MUST be able to indicate to the server to include only certain watchers (e.g. watchers from a particular domain, specific watchers defined by the filters) in the notifications.

6. Requirements for Uploading Rules (Operational Rules)

REQ C1: It MUST be possible for the client to upload the rules to the server and know the status - accepted or rejected - based on server policy.

REQ C2: Placing filtering rules in the body of the subscription MUST be supported.

REQ C3: The server MUST retain the uploaded filter setting for the duration of the subscription.

REQ C4: It MUST be possible to change the filter settings during a

subscription.

REQ C5: It MUST be possible for the client to reset the filter settings to the service (server) defined default.

REQ C6 It MUST be possible for a server not supporting filtering to inform the watcherinfo subscriber of the failure.

REQ C7: It MUST be possible for a server not understanding a filtering to inform the watcherinfo subscriber of the failure.

REQ C8: It MUST be possible for a server not accepting a filter to inform the watcherinfo subscriber of the reasons for not accepting the filter.

REQ C9: 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.

7. Security Considerations

Security requirements specified for [3] also applies to winfo filtering. Additional security considerations are described as follows.

REQ D1 It SHOULD be possible for the server to hide the fact that a filter was not acceptable.

REQ D2: The presence of a filter 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 watcherinfo subscriber to upload filters.

REQ D3 Modification to filters by an intermediary could also result in the watcherinfo subscriber either not receiving notifications of watcher information they are interested in or receiving a very large watcher info document. Therefore filters SHOULD be integrity protected between.

REQ D4: 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 D5: 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.

REQ D6: Authorization SHOULD occur irrespective of the filtering.

8. Example Applications for Notification Filtering

- o A presentity wishes to see who has subscribed to their presence. The presentity only wishes to see information for subscribers who are co-workers.
- o A presentity makes subscription to get information about active watchers.
- o A presentity makes subscription to get information about defined, new or unauthorised watchers.
- o A presentity requests information on watchers that have their status as "waiting", for authorization purposes.

9. Acknowledgements

The authors would like to thank Jonathan Rosenberg, Tim Moran, Juha Kalliokulju, Paul Kyzivat and Aki Niemi for their valuable input.

10. Changes from previous versions

10.1 Main changes from version 00

- o Added filtering model section.
- o Rephrased some requirements for clarity.
- o Rearranged requirements into more appropriate sections.

References

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

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

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

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

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

[6] Rosenberg, J., "An Extensible Markup Language (XML) Based Format for Watcher Information", draft-ietf-simple-winfo-format-04.txt, January 2003.

Authors' Addresses

Krisztian Kiss
Nokia
P.O. Box 100
33721 Tampere
Finland

Phone: + 358 50 4835363
EMail: krisztian.kiss@nokia.com

Eva Leppanen
Nokia
P.O BOX 785
Tampere
Finland

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

Hisham Khartabil
Nokia
P.O. Box 321
Helsinki
Finland

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

Intellectual Property Statement

Full Copyright Statement

Acknowledgment

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