A Conference List Information Event Package for the Session
Initiation Protocol (SIP)
draft-koren-sipping-conference-list-event-02.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

This document may only be posted in an Internet-Draft.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on July 3, 2008.

Abstract

This document describes the usage of the Session Initiation Protocol (SIP) for subscriptions and notifications related to conference lists.
A new conference list event package is specified. This event package allows a user to subscribe to a single event (the conference-list) and receive notifications that contain the list of conferences to which the user belongs and the status of each conference. The notifications sent from the conference server can contain either the entire list of the user's conferences or a partial list with the updates since the previous notification.

Table of Contents

1. Introduction.............................................2
2. Requirements Terminology...............................3
3. Usage Scenario...........................................3
4. Package Description....................................4
   4.1 Event package name....................................5
   4.2 Subscribe to conference-list event package..........5
   4.3 Subscription expiration................................5
   4.4 NOTIFY Bodies........................................5
5. XML Schema.............................................7
6. Example..................................................9
7. IANA Considerations....................................13
8. Security Considerations................................14
9. Acknowledgements.......................................14
10. Change History..........................................14
11. References.............................................14
   11.1 Normative References...............................14
   11.2 Informative References............................15
Author’s Addresses.........................................15
Intellectual Property Statement...........................16
Disclaimer of Validity.....................................16
Copyright Statement.......................................16
Acknowledgment...........................................17

1. Introduction

RFC 4575 [RFC4575] defines a SIP Event Package for Conference State. Users can subscribe to a conference Uniform Resource Identifier (URI) and be notified about their conference information.

The draft describes a new event package that allows a user to subscribe to a conference-list.

By subscribing to the event package, a user can receive a list of all relevant conference URIs (via the notification).

The user can then subscribe to each individual conference URI and receive specific conference information (according to [RFC4574]).

The user will receive updates (via notifications) whenever there is a change to the user’s conference list. Examples of such changes include the user joining a new conference, the user closing a conference, etc.

2. Requirements Terminology

Keywords "MUST", "MUST NOT", "REQUIRED", "SHOULD", "SHOULD NOT" and "MAY" that appear in this document are to be interpreted as described in [RFC2119].

3. Usage Scenario

Users subscribe to the conference-list event package in order to receive the most up-to-date information about their open and closed conferences. This information can be received on any device from which Users can subscribe. There are no restrictions regarding when a device can subscribe to the service. Users can decide to add additional UAs to subscribe to the event and get the most up-to-date recent conference list status. Furthermore, users can switch between different UAs and always get via notification the most updated conference list, which they can then use to subscribe to a specific conference Uniform Resource Identifier (URI) and get additional information (according to [RFC4575]).

As an example, User A can subscribe to the conference-list event package as follows:

```
SUBSCRIBE sip:A@example.com SIP/2.0
Via: SIP/2.0/UDP x12.example.com;branch=z9hG4bKv5as148
From: sip:A@example.com;tag=146sa4
To: sip:A@example.com
Call-ID: 553223@x12.example.com
Max-Forwards: 70
CSeq: 10 SUBSCRIBE
Contact: sip:A@x12.example.com
Expires: 3600
Event: conference-list
Accept: application/conference-list+xml
```
The conference server SHOULD check and authorize the subscription, and send the notification with the most updated conference list and status:

```
NOTIFY sip:A@example.com SIP/2.0
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:A@example.com;tag=151fd4
To: sip:A@example.com;tag=146sa4
Call-ID: 553223@x12.example.com
CSeq: 5 NOTIFY
Contact: sip:conference1.example.com
Event: conference-list
Subscription-State: active
Max-Forwards: 70
Content-Type: application/conference-list+xml
```

```
<?xml version="1.0"?><conference-list xmlns="urn:ietf:params:xml:ns:conference-list" version="1" state="full">
  <conferences resource="sip:A@example.com">
    <conference id="sip:conference_111@example.com" display-name="sip:conference_111@example.com" status="active"/>
  </conferences>
</conference-list>
```

4. Package Description

The Subscribe request with the conference-list event uses the same header and structure as defined in [RFC3265].
4.1 Event package name

A new event package is defined (as defined in [RFC3265]):
Event: conference-list

4.2 Subscribe to conference-list event package

A user MAY subscribe from any UA to the conference-list event package. The new event package SHOULD NOT include any body in the Subscription request. All other Subscribe operations (e.g. Refresh) are the same as defined in [RFC3265]. Subscription authentication MUST operate to validate the Subscriber. The Subscription authentication mechanism MAY operate as defined in [RFC3261]. Authorization of the subscribing user in the conference server is out of the scope of this draft.

4.3 Subscription expiration

The UA can subscribe with a defined expiration period to the conference server. The subscribing UA MAY use the Expires header field to determine the subscription period.

4.4 NOTIFY Bodies

The conference-list notification body is an XML document that MUST be well formed and SHOULD be valid. The conference-list XML document MUST be based on XML 1.0 and MUST be encoded by using UTF-8. The notify body structure and format is described by the XML schema in Section 5.
The notification from the conference server contains a new Content-Type:
Content-Type: application/conference-list+xml

The XML structure below describes the new conference-list Content-Type body:
<?xml version="1.0"?>
<conference-list xmlns="urn:ietf:params:xml:ns:conference-list" version="1" state="full">
  <conferences resource="sip:A@example.com">
    <conference id="sip:conference_111@example.com" display-name="sip:conference_111@example.com" status="active"/>
  </conferences>
</conference-list>

The conference list notification can include a full or partial list of URIs (in the state value). In case of state="full" the entire conference list MUST be sent from the conference server to the subscribed User. When state="partial", part of the conference list MUST be sent from the conference server to the subscribed user. The partial list SHOULD reflect the delta changes from the previous notification. The first notification to a subscribed user MUST be a full notification.

The "version" attribute is used to determine the sequence of notifications to the subscribed user. The conference server MUST increases the "version" for each notification within the same subscription.

The conference "id" describes the User’s unique conference Uniform Resource Identifier (URI). The "status" value defines the conference status - it may be "active" or "closed" (or any other status that describes the conference state). A conference that a User joins will have an "active" status. The status will be "closed" when a User closes an existing conference and a conference-list notification will be sent with the updated status value. "status" MUST be sent both in "partial" and "full" notifications. A "full" state will include only the conferences in "active" status while the "partial" state MAY include "active" and/or "closed" status. It is the responsibility of the UA of the subscriber to represent the complete conference list and status to the user.

The display-name value MAY be included in the conference tag. The display-name value MAY be a conference Uniform Resource Identifier
(URI) or other textual informative string. This information MAY be presented on the UI or may be used for any other purposes.

5. XML Schema

The following is the XML schema for the application/conference-list+xml data format. The notification body MUST follow below XML Schema format and structure:

```xml
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns="urn:ietf:params:xml:ns:conference_list"
  targetNamespace="urn:ietf:params:xml:ns:conference_list"
  elementFormDefault="qualified">
  <xs:element name="conference-list">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="conferences">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="conference" maxOccurs="unbounded">
                <xs:complexType>
                  <xs:attribute name="id" use="required">
                    <xs:simpleType>
                      <xs:restriction base="xs:string"/>
                    </xs:simpleType>
                  </xs:attribute>
                  <xs:attribute name="display-name" use="required">
                    <xs:simpleType>
                      <xs:restriction base="xs:string"/>
                    </xs:simpleType>
                  </xs:attribute>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```
<xs:attribute name="status" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="active"/>
      <xs:enumeration value="closed"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="resource" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string"/>
  </xs:simpleType>
</xs:attribute>
</xs:complexType>
</xs:element>
</xs:sequence>
<xs:attribute name="version" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:integer"/>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="state" use="required">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="partial"/>
      <xs:enumeration value="full"/>
    </xs:restriction>
  </xs:simpleType>
</xs:attribute>
6. Example

In the following example, Bob subscribes to the conference server in order to get the entire conference-list.

Bob SUBSCRIBE to conference-list

SUBSCRIBE sip:Bob@example.com SIP/2.0
Via: SIP/2.0/UDP x12.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=146sa4
To: sip:Bob@example.com
Call-ID: 553542@x12.example.com
Max-Forwards: 70
CSeq: 1 SUBSCRIBE
Contact: sip:Bob@x12.example.com
Event: conference-list
Accept: application/conference-list+xml
Expires: 3600

The conference server responds "200 OK" to the conference-list SUBSCRIBE request.

SIP/2.0 200 OK
Via: SIP/2.0/UDP x12.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=146sa4
To: sip:Bob@example.com;tag=1251fd
Call-ID: 553542@x12.example.com
CSeq: 1 SUBSCRIBE
Then the conference server sends a first full NOTIFY to the conference-list SUBSCRIBE.

```
NOTIFY sip:Bob@example.com SIP/2.0
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553542@x12.example.com
CSeq: 5 NOTIFY
Contact: sip:conference1.example.com
Event: conference-list
Subscription-State: active
Max-Forwards: 70
Content-Type: application/conference-list+xml
Content-Length: ...

<?xml version="1.0"?>
<conference-list xmlns="urn:ietf:params:xml:ns:conference-list" version="1" state="full">
  <conferences resource="sip:Bob@example.com">
    <conference id="sip:conference_111@example.com" display-name="sip:conference_111@example.com" status="active"/>
    <conference id="sip:conference_112@example.com" display-name="sip:conference_112@example.com" status="active"/>
    <conference id="sip:conference_113@example.com" display-name="sip:conference_113@example.com" status="active"/>
    <conference id="sip:conference_114@example.com" display-name="sip:conference_114@example.com" status="active"/>
  </conferences>
</conference-list>
```
Bob responds with "200 OK" to the NOTIFY:

SIP/2.0 200 OK
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553542@x12.example.com
CSeq: 5 NOTIFY

Partial NOTIFY to conference-list SUBSCRIBE due to changes in existing conferences (conference_111 & conference_114 closed) and joining a new conference (conference_115)

NOTIFY sip:Bob@x12.example.com SIP/2.0
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553542@x12.example.com
CSeq: 14 NOTIFY
Contact: sip:conference1.example.com
Event: conference-list
Subscription-State: active
Max-Forwards: 70
Content-Type: application/conference-list+xml
Content-Length:

<?xml version="1.0"?>
<conference-list xmlns="urn:ietf:params:xml:ns:conference-list" version="2" state="partial">
  <conferences resource="sip:Bob@example.com">
    <conference id="sip:conference_111@example.com" display-name="sip:conference_111@example.com" status="closed"/>
  </conferences>
</conference-list>
Bob responds with "200 OK" to the partial NOTIFY:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553542@x12.example.com
CSeq: 14 NOTIFY
```

Bob subscribes from another UA to conference-list and gets a new full notification from the conference server (with the most updated conference status):

```
NOTIFY sip:Bob@example.com SIP/2.0
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553635@x12.example.com
CSeq: 35 NOTIFY
Contact: sip:conference1.example.com
Event: conference-list
Subscription-State: active
Max-Forwards: 70
Content-Type: application/conference-list+xml
Content-Length: ...
```
<?xml version="1.0"?>
<conference-list xmlns="urn:ietf:params:xml:ns:conference-list"
version="1" state="full">
  <conferences resource="sip:Bob@example.com">
    <conference id="sip:conference_112@example.com" display-name="sip:conference_112@example.com" status="active"/>
    <conference id="sip:conference_113@example.com" display-name="sip:conference_113@example.com" status="active"/>
    <conference id="sip:conference_115@example.com" display-name="sip:conference_115@example.com" status="active"/>
  </conferences>
</conference-list>

Bob responds with "200 OK" to the NOTIFY:

SIP/2.0 200 OK
Via: SIP/2.0/UDP conference1.example.com;branch=z9hG4bKy5as148
From: sip:Bob@example.com;tag=1251fd
To: sip:Bob@example.com;tag=146sa4
Call-ID: 553635@x12.example.com
CSeq: 35 NOTIFY

7. IANA Considerations

The register specifications to this event are specified in RFC 3265
[RFC3265] Section 6.2.

Event Name: Conference-list

Template Package: No

Person to Contact: Oded Koren, oded.koren@comverse.com
8. Security Considerations

Security consideration follows those of presence subscriptions and Notification as detailed in RFC 3856 [RFC3856], sections 9.1, 9.2, 9.3

9. Acknowledgements

Thanks to Aloni Tali-Natali for the help and support.

10. Change History

Changes from draft-koren-sipping-conference-list-event-01:
Add sections 7 and 8
Add author

Changes from draft-koren-sipping-conference-list-event-00:
Changed "status" value from "close" to "closed".
Fixed syntax of examples.

11. References

11.1 Normative References


11.2 Informative References


Authors' Addresses

Oded Koren
Comverse
29 Habarzel St,
Tel Aviv 69710, Israel

Phone: +972 3 765 5395
Email: oded.koren@comverse.com
URI: http://www.comverse.com

Michael Fortinsky
Comverse
29 Habarzel St,
Tel Aviv 69710, Israel

Phone: +972 3 645 4029
Email: michael.fortinsky@comverse.com
URI: http://www.comverse.com

Ilan Ravid
Comverse
29 Habarzel St,
This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

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