Conference Establishment Using Request-Contained Lists in the Session Initiation Protocol (SIP)
draft-ietf-sipping-uri-list-conferencing-02.txt

Status of this Memo

   This document is an Internet-Draft and is subject to all provisions
   of section 3 of RFC 3667. 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 become aware will be disclosed, in accordance with
   RFC 3668.

   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 May 28, 2005.

Copyright Notice

Abstract

   This document describes how to create a conference using SIP URI-list
   services. In particular, we describe a mechanism that allows a
   client to provide a conference server with the initial list of
   participants using an INVITE-contained URI-list.

Table of Contents

[1](#). **Introduction**

   Section 4.5 of [3] describes how to create a conference using ad-hoc
   SIP [2] methods. The client sends an INVITE request to a conference
   factory URI, and receives the actual conference URI, which contains
   the "isfocus" feature tag, in the Contact header field of a response
   (typically a 200 OK).

   Once the client obtains the conference URI, it can add participants
   to the newly created conference in several ways, which are described
   in [3].

   Some environments have tough requirements regarding conference
   establishment time. So, they require the client to be able to
   request the creation of an ad-hoc conference and to provide the
   server with the initial set of participants in a single operation.
   This document describes how to meet this requirement using the
   mechanism to transport URI lists in SIP messages described in [4].

[2](#). **Terminology**

   In this document, the key words "MUST", "MUST NOT", "REQUIRED",
   "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT
   RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as
   described in BCP 14, RFC 2119 [1] and indicate requirement levels for
   compliant implementations.

[3](#). **Providing a Conference Server with a URI-List**

   A client that wants to include the set of initial participants in its
   initial INVITE to create an ad-hoc conference, adds a body whose
   disposition type is recipient-list, as defined in [4], with a
   URI-list that contains the participants that the client wants the
   server to INVITE. The client sends this INVITE to the conference
   factory URI.

[4](#). **URI List Format**

   As described in [4], specifications of individual URI-list services,
   like the conferencing service described here, need to specify a
   default format for recipient-list bodies used within the particular
   service.

   The default format for recipient-list bodies for conferencing UAs
   (User Agents) and servers is the resource list format defined in [5].
   So, conferencing UAs and servers handling recipient-list bodies MUST
   support this format and MAY support other formats.

Nevertheless, the Extensible Markup Language (XML) Configuration
Access Protocol (XCAP) resource list document provides features, such
as hierarchical lists and the ability to include entries by reference
relative to the XCAP root URI, that are not needed by the
conferencing service defined in this document, which only needs to
transfer a flat list of URIs between a UA and the conference server.
Therefore, when using the default resource list document,
conferencing UAs SHOULD use flat lists (i.e., no hierarchical lists)
and SHOULD NOT use <entry-ref> elements.

A conference factory application receiving a URI-list with more
information than what has just been described MAY discard all the
extra information.

Figure 1 shows an example of a flat list that follows the resource
list document.

```
<?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">
  <list>
    <entry uri="sip:bill@example.com" />
    <entry uri="sip:joe@example.org" />
    <entry uri="sip:ted@example.net" />
  </list>
</resource-lists>
```

                          Figure 1: URI List

5. Conference Server Behavior

On reception of an INVITE with a recipient-list body as described in
Section 3, a conference server MUST follow the rules described in [3]
to create ad-hoc conferences. Once the ad-hoc conference is created,
the conference server SHOULD attempt to add the participants in the
URI-list to the conference as if their addition had been requested
using any of the methods described in [3] (e.g., using CPCP [6]).

Once the conference server has created the ad-hoc conference and has
attempted to add the initial set of participants, the conference
server behaves as a regular conference server and MUST follow the
rules in [3].

Note that the status code in the response to the INVITE does not
provide any information about whether or not the conference server
was able to bring the users in the URI-list into the conference.

That is, a 200 (OK) means that the conference was created
successfully, that the client that generated the INVITE is in the
conference, and that the server understood the URI-list. If the
client wishes to obtain information about the status of other users
in the conference it SHOULD use general conference mechanisms, such
as the conference package [7].

6. Re-INVITEs

The previous Sections have specified how to include a URI-list in an
initial INVITE request to a conference server. Once the
INVITE-initiated dialog between the client and the conference server
has been established, the client may need to send subsequent INVITE
requests (typically referred to as re-INVITEs) to the conference
server to, for example, modify the characteristics of the media
exchanged with the server.

At this point, there are no semantics associated with resource-list
bodies in re-INVITEs (although future extensions may define them).
Therefore, clients SHOULD NOT include resource-list bodies in
re-INVITEs sent to a conference server.

A conference server receiving a re-INVITE with a resource-list body,
following standard SIP procedures, rejects it with a 415 (Unsupported
Media Type) response.

   Note that a difference between an initial INVITE request and a
   re-INVITE is that while the initial INVITE is sent to the
   conference factory URI, the re-INVITE is sent to the URI provided
   by the server in a Contact header field when the dialog was
   established. Therefore, from the client's point of view, the
   resource identified by the former URI supports recipient-list
   bodies while the resource identified by the latter does not
   support them.

7. Example

The following is an example of an INVITE request, which carries a URI
list in a recipient-list body part, sent by a UA to a conference
factory application. Note that since the INVITE carries an SDP
description as well, it contains a multipart body.

   INVITE sip:conf-fact@example.com SIP/2.0
   Via: SIP/2.0/TCP client.chicago.example.com
       ;branch=z9hG4bKhjhs8ass83
   Max-Forwards: 70
   To: Conf Factory <sip:conf-fact@example.com>

```
   From: Carol <sip:carol@chicago.example.com>;tag=32331
   Call-ID: d432fa84b4c76e66710
   CSeq: 1 INVITE
   Contact: <sip:carol@client.chicago.example.com>
   Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER,
         SUBSCRIBE, NOTIFY
   Allow-Events: dialog
   Accept: application/sdp, message/sipfrag
   Conten-Type: multipart/mixed;boundary="boundary1"
   Content-Length: 690

   --boundary1
   Content-Type: application/sdp

   v=0
   o=carol 2890844526 2890842807 IN IP4 chicago.example.com
   s=-
   c=IN IP4 192.0.2.1
   t=0 0
   m=audio 20000 RTP/AVP 0
   a=rtpmap:0 PCMU/8000
   m=video 20002 RTP/AVP 31
   a=rtpmap:31 H261/90000

   --boundary1
   Content-Type: application/resource-lists+xml
   Content-Disposition: recipient-list

   <?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">
     <list>
       <entry uri="sip:bill@example.com" />
       <entry uri="sip:joe@example.org" />
       <entry uri="sip:ted@example.net" />
     </list>
   </resource-lists>
   --boundary1--
```

                        Figure 2: INVITE request

8. **Security Considerations**

   This document discusses setup of SIP conferences using a
   request-contained URI-list. Both conferencing and URI-lists services
   have specific security requirements which will be summarized here.
   Conferences generally have authorization rules about who may or may

not join a conference, what type of media may or may not be used, etc. This information is used by the focus to admit or deny participation in a conference. It is RECOMMENDED that these types of authorization rules be used to provide security for a SIP conference.

For this authorization information to be used, the focus needs to be able to authenticate potential participants. Normal SIP mechanisms including Digest authentication and certificates can be used. These conference specific security requirements are discussed further in the requirements and framework documents.

For conference creation using a list, there are some additional security considerations. The Framework and Security Considerations for SIP URI-List Services [4] discusses issues related to SIP URI-list services. Given that a conference server sending INVITEs to a set of users acts as an URI-list service, implementations of conference servers that handle lists MUST follow the security-related rules in [4]. These rules include mandatory authentication and authorization of clients, and opt-in lists.

9. Acknowledges

Cullen Jennings, Hisham Khartabil, and Jonathan Rosenberg provided useful comments on this document.

10. References

10.1 Normative References

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

[2]   Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[3]   Johnston, A. and O. Levin, "Session Initiation Protocol Call Control - Conferencing for User Agents", draft-ietf-sipping-cc-conferencing-05 (work in progress), October 2004.

[4]   Camarillo, G., "Requirements and Framework for Session Initiation Protocol (SIP)Uniform Resource Identifier (URI)-List Services", draft-ietf-sipping-uri-services-01 (work in progress), October 2004.

[5]   Rosenberg, J., "Extensible Markup Language (XML) Formats for Representing Resource Lists",

draft-ietf-simple-xcap-list-usage-04 (work in progress), October
2004.

10.2 Informational References

[6]   Koskelainen, P. and H. Khartabil, "An Extensible Markup Language
      (XML) Configuration Access Protocol (XCAP) Usage for Conference
      Policy Manipulation", draft-koskelainen-xcon-xcap-cpcp-usage-02
      (work in progress), February 2004.

[7]   Rosenberg, J. and H. Schulzrinne, "A Session Initiation Protocol
      (SIP) Event Package for Conference State",
      draft-ietf-sipping-conference-package-06 (work in progress),
      October 2004.

Authors' Addresses

   Gonzalo Camarillo
   Ericsson
   Hirsalantie 11
   Jorvas 02420
   Finland

   EMail: Gonzalo.Camarillo@ericsson.com

   Alan Johnston
   MCI
   100 South 4th Street
   St. Louis, MO 63102
   USA

   EMail: alan.johnston@mci.com

Intellectual Property Statement