Referring to Multiple Resources in the Session Initiation Protocol (SIP)

Status of This Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

Abstract

   This document defines extensions to the SIP REFER method so that it
   can be used to refer to multiple resources in a single request.
   These extensions include the use of pointers to Uniform Resource
   Identifier (URI) lists in the Refer-To header field and the
   "multiple-refer" SIP option-tag.

Table of Contents

1. Introduction

   RFC 3261 (SIP) [RFC3261] is extended by RFC 3515 [RFC3515] with a
   REFER method that allows a user agent (UA) to request a second UA to
   send a SIP request to a third party. For example, if Alice is in a
   call with Bob, and decides Bob needs to talk to Carol, Alice can
   instruct her SIP UA to send a REFER request to Bob's UA providing
   Carol's SIP Contact information. Assuming Bob has given it
   permission, Bob's UA will attempt to call Carol using that contact.
   That is, it will send an INVITE request to that contact.

   A number of applications need to request this second UA to initiate
   transactions towards a set of destinations. In one example, the
   moderator of a conference may want the conference server to send BYE
   requests to a group of participants. In another example, the same
   moderator may want the conference server to INVITE a set of new
   participants.

   We define an extension to the REFER method so that REFER requests can
   be used to refer other user agents (such as conference servers) to
   multiple destinations. In addition, this mechanism uses the
   suppression of the REFER method implicit subscription specified in
   RFC 4488 [RFC4488].

2. Terminology

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in BCP 14, RFC 2119
   [RFC2119] and indicate requirement levels for compliant
   implementations.

   This document reuses the following terminology defined in RFC 3261
   [RFC3261]:

   o   User Agent (UA)
   o   User Agent Client (UAC)
   o   User Agent Server (UAS)

   This document defines the following new terms:

   REFER-Issuer: a user agent issuing a REFER request.

   REFER-Recipient: an entity receiving a REFER request and forwarding
      a SIP request to a number of REFER-Targets. The REFER-Recipient
      is typically a network entity, such as a URI-list server, that
      acts as a UAS for REFER requests and as a UAC for other SIP
      requests.

REFER-Target: a UA of the intended final recipient of a SIP request
   generated by the REFER-Recipient.

3. Overview of Operation

   This document describes an application of URI-list services [RFC5363]
   that allows a URI-list service to receive a SIP REFER request
   containing a list of targets. The URI-list service invokes the
   requested SIP method to each of the targets contained in the list.
   This type of URI-list service is referred to as a REFER-Recipient
   throughout this document.

   This document defines an extension to the SIP REFER method specified
   in RFC 3515 [RFC3515] that allows a SIP UAC to include a URI list as
   specified in RFC 4826 [RFC4826] of REFER-Targets in a REFER request
   and send it to a REFER-Recipient. The REFER-Recipient creates a new
   SIP request for each entry in the URI list and sends it to each
   REFER-Recipient.

   The URI list that contains the list of targets is used in conjunction
   with RFC 5364 [RFC5364] to allow the sender indicate the role (e.g.,
   'to', 'cc', or anonymous) in which the REFER-Target is involved in
   the signalling.

   We represent multiple targets of a REFER request using a URI list as
   specified in RFC 4826 [RFC4826]. A REFER-Issuer that wants to refer
   a REFER-Recipient to a set of destinations creates a SIP REFER
   request. The Refer-To header contains a pointer to a URI list, which
   is included in a body part, and an option-tag in the Require header
   field: "multiple-refer". This option-tag indicates the requirement
   to support the functionality described in this specification.

   When the REFER-Recipient receives such a request, it creates a new
   request per REFER-Target and sends them, one to each REFER-Target.

   This document does not provide any mechanism for REFER-Issuers to
   find out about the results of a REFER request containing multiple
   REFER-Targets. Furthermore, it does not provide support for the
   implicit subscription mechanism that is part of the SIP REFER method.
   The way REFER-Issuers are kept informed about the results of a REFER
   is service specific. For example, a REFER-Issuer sending a REFER
   request to invite a set of participants to a conference can discover
   which participants were successfully brought into the conference by
   subscribing to the conference state event package specified in RFC
   4575 [RFC4575].

4. The multiple-refer SIP Option-Tag

   We define a new SIP option-tag for the Require and Supported header
   fields: "multiple-refer".

   A user agent including the "multiple-refer" option-tag in a Supported
   header field indicates compliance with this specification.

   A user agent generating a REFER with a pointer to a URI list in its
   Refer-To header field MUST include the "multiple-refer" option-tag in
   the Require header field of the REFER.

5. Suppressing REFER's Implicit Subscription

   REFER requests with a single REFER-Target establish implicitly a
   subscription to the refer event. The REFER-Issuer is informed about
   the result of the transaction towards the REFER-Target through this
   implicit subscription. As described in RFC 3515 [RFC3515], NOTIFY
   requests sent as a result of an implicit subscription created by a
   REFER request contain a body of type "message/sipfrag", RFC 3420
   [RFC3420], that describes the status of the transaction initiated by
   the REFER-Recipient.

   In the case of a REFER-Issuer that generates a REFER with multiple
   REFER-targets, the REFER-Issuer is typically already subscribed to
   other event packages that can provide the information about the
   result of the transactions towards the REFER-Targets. For example, a
   moderator instructing a conference server to send a BYE request to a
   set of participants is usually subscribed to the conference state
   event package for the conference. Notifications to this event
   package will keep the moderator and the rest of the subscribers
   informed of the current list of conference participants.

   Most of the applications using the multiple REFER technology
   described in this memo do not need its implicit subscription.
   Consequently, a SIP REFER-Issuer generating a REFER request with
   multiple REFER-Targets SHOULD include the "norefersub" option-tag in
   a Require header field and SHOULD include a Refer-Sub header field
   set to "false" to indicate that no notifications about the requests
   should be sent to the REFER-Issuer. The REFER-Recipient SHOULD honor
   the suggestion and also include a Refer-Sub header field set to
   "false" in the 200 (OK) response. The "norefersub" SIP option-tag
   and the Refer-Sub header field are specified in RFC 4488 [RFC4488].

      RFC 4488 [RFC4488] indicates that a condition for the REFER-Issuer
      to include a Refer-Sub header is that the REFER-Issuer is sure
      that the REFER request will not fork.

At the time of writing, there is no extension that allows to report
the status of several transactions over the implicit subscription
associated with a REFER dialog. That is the motivation for this
document to recommend the usage of the "norefersub" option-tag. If
in the future such an extension is defined, REFER-Issuers using it
could refrain from using the "norefersub" option-tag and use the new
extension instead.

6. URI-List Format

   As described in RFC 5363 [RFC5363], specifications of individual URI-
   list services need to specify a default format for 'recipient-list'
   bodies used within the particular service.

   The default format for 'recipient-list' bodies for REFER-Issuers and
   REFER-Recipients is RFC 4826 [RFC4826] extended with RFC 5364
   [RFC5364]. REFER-Recipients handling 'recipient-list' bodies MUST
   support both of these formats. Both REFER-Issuers and REFER-
   Recipients MAY support other formats.

   As described in RFC 5364 [RFC5364], each URI can be tagged with a
   'copyControl' attribute set to either "to", "cc", or "bcc",
   indicating the role in which the target will get the referred SIP
   request. However, depending on the target SIP method, a
   'copyControl' attribute lacks sense. For example, while a
   'copyControl' attribute can be applied to INVITE requests, it does
   not make sense with mid-dialog requests such as BYE requests.

   In addition to the 'copyControl' attribute, URIs can be tagged with
   the 'anonymize' attribute (also specified in RFC 5364 [RFC5364]) to
   prevent that the REFER-Recipient discloses the target URI in a URI
   list.

   Additionally, RFC 5364 [RFC5364] defines a 'recipient-list-history'
   body that contains the list of targets. The default format for
   'recipient-list-history' bodies for conference services is also RFC
   4826 [RFC4826] extended with RFC 5364 [RFC5364]. REFER-Recipients
   supporting this specification MUST support both of these formats;
   REFER-Targets MAY support these formats. Both REFER-Recipients and
   REFER-Targets MAY support other formats.

   Nevertheless, RFC 4826 [RFC4826] provides features, such as
   hierarchical lists and the ability to include entries by reference
   relative to the XML Configuration Access Protocol (XCAP) root URI,
   that are not needed by the multiple REFER service defined in this
   document.

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:cp="urn:ietf:params:xml:ns:copycontrol">

  <list>
    <entry uri="sip:bill@example.com" cp:copyControl="to"  />
    <entry uri="sip:joe@example.org" cp:copyControl="cc" />
    <entry uri="sip:ted@example.net" cp:copyControl="bcc" />
  </list>
</resource-lists>
```

Figure 1: URI list

7. Behavior of SIP REFER-Issuers

   As indicated in Sections 4 and 5, a SIP REFER-Issuer that creates a
   REFER request with multiple REFER-Targets includes a "multiple-refer"
   and "norefersub" option-tags in the Require header field and, if
   appropriate, a Refer-Sub header field set to "false". The REFER-
   Issuer includes the set of REFER-Targets in a recipient-list body
   whose disposition type is 'recipient-list', as defined in RFC 5363
   [RFC5363]. The URI-list body is further described in Section 6.

   The Refer-To header field of a REFER request with multiple REFER-
   Targets MUST contain a pointer (i.e., a Content-ID Uniform Resource
   Locator (URL) as per RFC 2392 [RFC2392]) that points to the body part
   that carries the URI list. The REFER-Issuer SHOULD NOT include any
   particular URI more than once in the URI list.

   RFC 4826 [RFC4826] provides features, such as hierarchical lists and
   the ability to include entries by reference relative to the XCAP root
   URI. However, these features are not needed by the multiple REFER
   service defined in this document. Therefore, when using the default
   resource list document, SIP REFER-Issuers generating REFER requests
   with multiple REFER-Targets SHOULD use flat lists (i.e., no
   hierarchical lists) and SHOULD NOT use <entry-ref> elements.

8. Behavior of REFER-Recipients

   The REFER-Recipient follows the rules in Section 2.4.2 of RFC 3515
   [RFC3515] to determine the status code of the response to the REFER.

   The REFER-Recipient SHOULD not create an implicit subscription, and
   SHOULD add a Refer-Sub header field set to "false" in the 200 OK
   response.

The incoming REFER request typically contains a URI-list document or reference with the actual list of targets. If this URI list includes resources tagged with the 'copyControl' attribute set to a value of "to" or "cc", and if the request is appropriate for the service, e.g., it is not received mid-dialog, the REFER-Recipient SHOULD include a URI list in each of the outgoing requests. This list SHOULD be formatted according to RFC 4826 [RFC4826] and RFC 5364 [RFC5364]. The REFER-Recipient MUST follow the procedures specified in RFC 4826 [RFC4826] with respect to handling of the 'anonymize', 'count', and 'copyControl' attributes.

Section 4 of RFC 5363 [RFC5363] discusses cases when duplicated URIs are found in a URI list. In order to avoid duplicated requests, REFER-Recipients MUST take those actions specified in RFC 5363 [RFC5363] into account to avoid sending a duplicated request to the same target.

If the REFER-Recipient includes a URI list in an outgoing request, it MUST include a Content-Disposition header field, specified in RFC 2183 [RFC2183], with the value set to 'recipient-list-history' and a 'handling' parameter, specified in RFC 3204 [RFC3204], set to "optional".

Since the multiple REFER service does not use hierarchical lists nor lists that include entries by reference to the XCAP root URI, a REFER-Recipient receiving a URI list with more information than what has been described in Section 6 MAY discard all the extra information.

The REFER-Recipient follows the rules in RFC 3515 [RFC3515] to generate the necessary requests towards the REFER-Targets, acting as if it had received a regular (no URI list) REFER per each URI in the URI list.

9. Example

Figure 2 shows an example flow where a REFER-Issuer sends a multiple-REFER request to the focus of a conference, which acts as the REFER-Recipient. The REFER-Recipient generates a BYE request per REFER-Target. Details for using REFER request to remove participants from a conference are specified in RFC 4579 [RFC4579].

```
   +--------+     +---------+    +--------+  +--------+  +--------+
   | REFER  |     | REFER   |    | REFER  |  | REFER  |  | REFER  |
   | issuer |     |recipient|    |target 1|  |target 2|  |target 3|
   |        |     |         |    |        |  |        |  |        |
   | Carol  |     | (focus) |    | Bill   |  | Joe    |  | Ted    |
   +--------+     +---------+    +--------+  +--------+  +--------+
        | 1. REFER       |           |           |           |
        | -------------->|           |           |           |
        | 2. 202 Accepted|           |           |           |
        |<-------------- |  3. BYE   |           |           |
        |                | --------->|           |           |
        |                |  4. BYE   |           |           |
        |                | --------------------->|           |
        |                |  5. BYE   |           |           |
        |                | --------------------------------->|
        |                |  6. 200 OK|           |           |
        |                |<--------- |           |           |
        |                |  7. 200 OK|           |           |
        |                |<--------------------- |           |
        |                |  8. 200 OK|           |           |
        |                |<--------------------------------- |
        |                |           |           |           |
        |                |           |           |           |
        |                |           |           |           |
        |                |           |           |           |
```
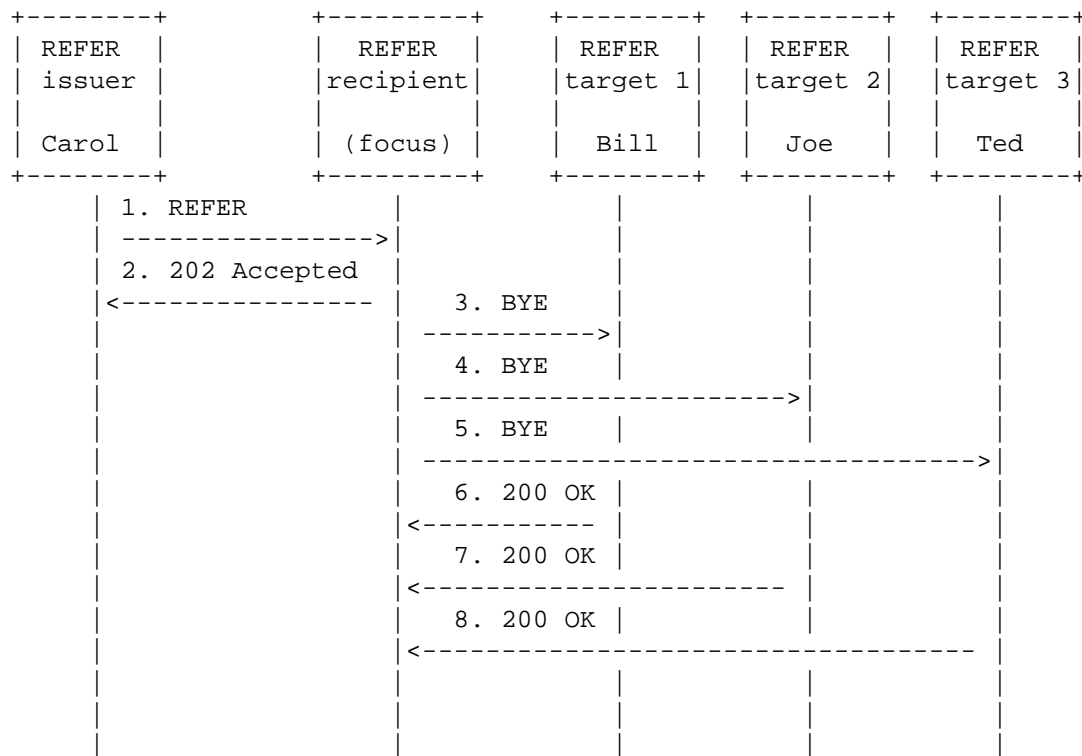
           Figure 2: Example flow of a REFER request containing
                        multiple REFER-Targets

   The REFER request (1) contains a Refer-To header field that includes
   a pointer to the message body, which carries a list with the URIs of
   the REFER-Targets. In this example, the URI list does not contain
   the 'copyControl' attribute extension. The REFER's Require header
   field carries the "multiple-refer" and "norefersub" option-tags. The
   Request-URI is set to a Globally Routable User Agent URI (GRUU)
   [SIP-GRUU] (as a guarantee that the REFER request will not fork).
   The Refer-Sub header field is set to "false" to request the
   suppression of the implicit subscription. Figure 3 shows an example
   of this REFER request. The resource list document contains the list
   of REFER-Target URIs along with the method of the SIP request that
   the REFER-Recipient generates.

```
REFER sip:conf-123@example.com;gruu;opaque=hha9s8d-999a  SIP/2.0
Via: SIP/2.0/TCP client.chicago.example.com
        ;branch=z9hG4bKhjhs8ass83
Max-Forwards: 70
To: "Conference 123" <sip:conf-123@example.com>
From: Carol <sip:carol@chicago.example.com>;tag=32331
Call-ID: d432fa84b4c76e66710
CSeq: 2 REFER
Contact: <sip:carol@client.chicago.example.com>
Refer-To: <cid:cn35t8jf02@example.com>
Refer-Sub: false
Require: multiple-refer, norefersub
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
Allow-Events: dialog
Accept: application/sdp, message/sipfrag
Content-Type: application/resource-lists+xml
Content-Disposition: recipient-list
Content-Length: 362
Content-ID: <cn35t8jf02@example.com>

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

           Figure 3: REFER request with multiple REFER-Targets

Figure 4 shows an example of the BYE request (3) that the REFER-
Recipient sends to the first REFER-Target.

```
BYE sip:bill@example.com SIP/2.0
Via: SIP/2.0/TCP conference.example.com
        ;branch=z9hG4bKhjhs8assmm
Max-Forwards: 70
From: "Conference 123" <sip:conf-123@example.com>;tag=88734
To: <sip:bill@example.com>;tag=29872
Call-ID: d432fa84b4c34098s812
CSeq: 34 BYE
Content-Length: 0
```

                          Figure 4: BYE request

10. Security Considerations

   RFC 5363 [RFC5363] discusses issues related to SIP URI-list services.
   Given that a REFER-Recipient accepting REFER requests with multiple
   REFER-targets acts as a URI-list service, implementations of this
   type of server MUST follow the security-related rules in RFC 5363
   [RFC5363]. These rules include opt-in lists and mandatory
   authentication and authorization of clients.

   Additionally, REFER-Recipients SHOULD only accept REFER requests
   within the context of an application that the REFER-Recipient
   understands (e.g., a conferencing application). This implies that
   REFER-Recipients MUST NOT accept REFER requests for methods they do
   not understand. The idea behind these two rules is that REFER-
   Recipients are not used as dumb servers whose only function is to
   fan-out random messages they do not understand.

11. IANA Considerations

   This document defines a new SIP option-tag: "multiple-refer". This
   option-tag has been registered in the SIP Parameters registry.

   The following row has been added to the "Option Tags" section of the
   SIP Parameter Registry:

   +-----------------+-----------------------------------+-----------+
   | Name | Description | Reference |
   +-----------------+-----------------------------------+-----------+
multiple-refer	This option tag indicates support	[RFC5368]
	for REFER requests that contain a	
	resource list document describing	
	multiple REFER targets.	
   +-----------------+-----------------------------------+-----------+

       Table 1: Registration of the 'multiple-refer' option-tag in SIP

12. References

12.1. Normative References

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
                Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC2183]   Troost, R., Dorner, S., and K. Moore, "Communicating
                Presentation Information in Internet Messages: The
                Content-Disposition Header Field", RFC 2183, August 1997.

   [RFC2392]   Levinson, E., "Content-ID and Message-ID Uniform Resource
               Locators", RFC 2392, August 1998.

   [RFC3204]   Zimmerer, E., Peterson, J., Vemuri, A., Ong, L., Audet,
               F., Watson, M., and M. Zonoun, "MIME media types for ISUP
               and QSIG Objects", RFC 3204, December 2001.

   [RFC3261]   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.

   [RFC3420]   Sparks, R., "Internet Media Type message/sipfrag",
               RFC 3420, November 2002.

   [RFC3515]   Sparks, R., "The Session Initiation Protocol (SIP) Refer
               Method", RFC 3515, April 2003.

   [RFC4488]   Levin, O., "Suppression of Session Initiation Protocol
               (SIP) REFER Method Implicit Subscription", RFC 4488,
               May 2006.

   [RFC4826]   Rosenberg, J., "Extensible Markup Language (XML) Formats
               for Representing Resource Lists", RFC 4826, May 2007.

   [RFC5363]   Camarillo, G. and A.B. Roach, "Framework and Security
               Considerations for Session Initiation Protocol (SIP) URI-
               List Services", RFC 5363, October 2008.

   [RFC5364]   Garcia-Martin, M. and G. Camarillo, "Extensible Markup
               Language (XML) Format Extension for Representing Copy
               Control Attributes in Resource Lists", RFC 5364,
               October 2008.

12.2. Informative References

   [RFC4575]   Rosenberg, J., Schulzrinne, H., and O. Levin, "A Session
               Initiation Protocol (SIP) Event Package for Conference
               State", RFC 4575, August 2006.

   [RFC4579]   Johnston, A. and O. Levin, "Session Initiation Protocol
               (SIP) Call Control - Conferencing for User Agents",
               BCP 119, RFC 4579, August 2006.

   [SIP-GRUU] Rosenberg, J., "Obtaining and Using Globally Routable
               User Agent (UA) URIs (GRUU) in the Session Initiation
               Protocol (SIP)", Work in Progress, October 2007.

Authors' Addresses

   Gonzalo Camarillo
   Ericsson
   Hirsalantie 11
   Jorvas 02420
   Finland

   EMail: Gonzalo.Camarillo@ericsson.com

   Aki Niemi
   Nokia
   P.O. Box 321
   NOKIA GROUP, FIN 00045
   Finland

   EMail: Aki.Niemi@nokia.com

   Markus Isomaki
   Nokia
   P.O. Box 100
   NOKIA GROUP, FIN 00045
   Finland

   EMail: markus.isomaki@nokia.com

   Miguel A. Garcia-Martin
   Ericsson
   Via de los Poblados 13
   Madrid 28033
   Spain

   EMail: miguel.a.garcia@ericsson.com

   Hisham Khartabil
   Ericsson Australia

   EMail: hisham.khartabil@gmail.com