

SIPPING Working Group
Internet-Draft
Expires: November 12, 2006

G. Camarillo
Ericsson
A. Roach
Estacado Systems
O. Levin
Microsoft Corporation
May 11, 2006

**Subscriptions to Request-Contained Resource Lists in the Session
Initiation Protocol (SIP)
draft-ietf-sipping-uri-list-subscribe-05.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](#).

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 November 12, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006).

Abstract

This document specifies a way to create subscription to a list of resources in SIP. This is achieved by including the list of resources in the body of a SUBSCRIBE. Instead of having a subscriber send a SUBSCRIBE for each resource individually, the subscriber

defines the resource list, subscribes to it, and gets notifications about changes in the resources' state using a single SUBSCRIBE dialog.

Table of Contents

1.	Introduction	3
2.	Terminology	3
3.	Providing a Resource List Server with a URI-List	3
4.	URI-List Format	3
5.	Resource List Server Behavior	4
6.	Subsequent SUBSCRIBEs	4
7.	Option-tag	5
8.	Providing a URI to Manipulate a Resource List	5
9.	Example	6
10.	Security Considerations	7
11.	IANA Considerations	7
12.	Acknowledges	7
13.	Normative References	7
	Authors' Addresses	9
	Intellectual Property and Copyright Statements	10

1. Introduction

RFC xxxx [4] specifies how to establish subscriptions to a homogeneous resource list in SIP [2] and defines the procedures for getting notifications about changes in the state of the associated resources. Yet, list creation is outside the scope of [4].

This document specifies a way to create a list with a set of resources and subscribe to it using a single SIP request. This is achieved by including the list of resources (as defined in [5]) in the body of the SUBSCRIBE request.

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 Resource List Server with a URI-List

A client that wants to create a resource list and subscribe to it, using the mechanism described in this document, constructs a SUBSCRIBE with at least one body, whose disposition is type "recipient-list" as defined in [5], containing the URI-list. The client MUST build the remaining of the SUBSCRIBE request following the rules in RFC 3265 [3].

The client MUST support the "rlmi+xml" format defined in [4] and signal this by including "rlmi+xml" in the Accept header. The client MAY support additional formats and include them in the Accept header field of the SUBSCRIBE.

4. URI-List Format

The [5] mandates that each URI-list services specification, such as the subscription service defined here, specifies the default format for the recipient-list bodies used within the particular service.

The default format for the recipient-list bodies for the subscription service defined in this document is the resource list format defined in [6]. UAs (User Agents) and resource list servers handling recipient-list bodies MUST support this format and MAY support other formats.

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 subscription service defined here, which only needs to transfer a flat list of URIs between a UA and the resource list server. Therefore, when using the default resource list document, UAs SHOULD use flat lists (i.e., no hierarchical lists) and SHOULD NOT use <entry-ref> elements.

A resource list server 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. Resource List Server Behavior

On reception of a SUBSCRIBE with a URI-list, a resource list server, which chooses to accept the "rlmi+xml" format, MUST comply with [4] for creating the subscription and reporting the changes in the resources within the created dialog.

Note that the status code in the response to the SUBSCRIBE does not provide any information about whether or not the resource list server was able to successfully subscribe to the URIs in the URI-list. The client obtains this information in the notifications sent by the server.

6. Subsequent SUBSCRIBES

The previous Sections have specified how to include a URI-list in an initial SUBSCRIBE request to a resource list server in order to subscribe to the state of a set of resources. Once the subscription

has been created and a dialog between the client and the resource list server has been established, the client may need to send subsequent SUBSCRIBE requests to, for example, extend the duration of the subscription.

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

A resource list server receiving a subsequent SUBSCRIBE request 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 SUBSCRIBE request and subsequent ones is that while the initial request is sent to the public URI of the resource list, subsequent ones are sent to the URI provided by the server 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. Option-tag

This document defines the 'recipient-list-subscribe' option-tag for use in the Require and Supported SIP header fields.

User agent clients generating a SUBSCRIBE with a recipient-list body, as described in previous sections, MUST include this option-tag in a Require header field. User agents that are able to receive and process SUBSCRIBEs with a recipient-list body, as described in previous sections, SHOULD include this option-tag in a Supported header field when responding to OPTIONS requests.

8. Providing a URI to Manipulate a Resource List

A client may need to manipulate a resource list at a resource list server. The resource list server MAY provide a URI to manipulate the resource list associated with a subscription using the Call-Info header field in the NOTIFY that establishes the subscription. The "purpose" parameter of the Call-Info header field MUST have a value of 'list-management', which we register with the IANA in [Section 11](#). The following is an example of such a header field.

Call-Info: <http://xcap.example.com/your-list.xml>

;purpose=list-management

The life-time of a resource list to be manipulated by the URI provided by the server is blundled to the life-time of the subscription. That is, the resource list SHOULD be destroyed when the subscription expires or is otherwise terminated.

9. Example

The following is an example of a SUBSCRIBE request, which carries a URI-list in its body, sent by a UA to a resource list server.

```
SUBSCRIBE sip:rls@example.com SIP/2.0
Via: SIP/2.0/TCP terminal.example.com;branch=z9hG4bKwYb6QREiCL
Max-Forwards: 70
To: RLS <sip:rls@example.com>
From: <sip:adam@example.com>;tag=ie4hbb8t
Call-ID: cdB34qLToc@terminal.example.com
CSeq: 1 SUBSCRIBE
Contact: <sip:terminal.example.com>
Event: presence
Expires: 7200
Require: recipient-list-subscribe
Supported: eventlist
Accept: application/cpim-pidf+xml
Accept: application/rlmi+xml
Accept: multipart/related
Accept: multipart/signed
Accept: multipart/encrypted
Content-Type: application/resource-lists+xml
Content-Disposition: recipient-list
Content-Length: 337

<?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 2: SUBSCRIBE request

10. Security Considerations

The Security Considerations Section of [\[4\]](#) discusses security issues related to resource list servers. Resource list servers accepting request-contained URI-lists MUST also follow the security guidelines given in [\[4\]](#).

The Framework and Security Considerations for SIP URI-List Services [\[5\]](#) discusses issues related to SIP URI-list services. Given that a resource list server sending SUBSCRIBE to a set of users acts as a URI-list service, implementations of resource list servers that handle request-contained URI-lists MUST follow the security-related rules in [\[5\]](#). These rules include mandatory authentication and authorization of clients, and opt-in lists.

11. IANA Considerations

The document defines the 'list-management' value for the purpose parameter of the Call-Info header field. A reference to this RFC (in double brackets) needs to be added to the purpose Call-Info parameter entry in the SIP Parameters registry.

This document defines the 'recipient-list-subscribe' SIP option-tag in [Section 7](#). It should be registered in the Option Tags subregistry under the SIP parameter registry. The following is the description to be used in the registration.

This option-tag is used to ensure that a server can process the 'recipient-list' body used in a SUBSCRIBE request.

12. Acknowledges

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

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

- [4] Roach, A., Rosenberg, J., and B. Campbell, "A Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists", [draft-ietf-simple-event-list-07](#) (work in progress), January 2005.
- [5] Camarillo, G. and A. Roach, "Framework and Security Considerations for Session Initiation Protocol (SIP) Uniform Resource Identifier (URI)-List Services", [draft-ietf-sipping-uri-services-05](#) (work in progress), January 2006.
- [6] Rosenberg, J., "Extensible Markup Language (XML) Formats for Representing Resource Lists", [draft-ietf-simple-xcap-list-usage-05](#) (work in progress), February 2005.

Authors' Addresses

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Gonzalo.Camarillo@ericsson.com

Adam Roach
Estacado Systems

Email: adam@estacado.net

Orit Levin
Microsoft Corporation
One Microsoft Way
Redmond, WA 98052

Email: oritl@microsoft.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2006). 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.

