Response Code for Dynamic Proxy Redirect
draft-rajesh-sipping-303-01

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 August 31, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2007).

Abstract

The Session Initiation Protocol (SIP) allows for sessions to be redirected by the use of the 3xx response code. This response code allows redirect servers and proxies to redirect the session or for the message to be sent all the way back to the initiating client to recurse. There is no way to direct the domain responsible for the Request-URI to perform the redirection. This document defines a new response code that clients and redirect servers could use to ensure
that recursion take place within the domain responsible for the Request-URI rather than allowing the initiating client do perform the redirect.

Requirements Language

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 RFC 2119 [RFC2119].

Table of Contents

1. Introduction .................................................. 3
2. Server Behavior .............................................. 3
   2.1. User Agent Servers ....................................... 3
   2.2. Redirection Servers ...................................... 4
3. Proxy Behavior .................................................. 4
   3.1. Example Flow ............................................. 5
4. Client Behavior .................................................. 5
5. 303 Proxy Redirect ............................................ 6
6. IANA Considerations ........................................... 6
7. Security Considerations ........................................ 6
8. Changes from 00 ................................................ 6
9. Acknowledgements .............................................. 7
10. Normative References ......................................... 7
Authors’ Addresses .................................................. 7
Intellectual Property and Copyright Statements .................. 8
1. Introduction

The Session Initiation Protocol (SIP) (1) allows for sessions to be redirected by the use of the 3xx response code. The specification allows for the use of Redirect servers to push redirection information back to the client, similarly UAS’s may send back one or more alternate locations by sending back URI(s) in the Contact header. The clients upon receiving this response will build a target set and send a new request to the URI(s) in the set.

However there are many situations when the redirect server or UAS may not want the 3xx to be sent all the way back to the originating client, and the call to be recursed by a proxy within its domain. Some examples could include internally routable SIP URIs, restricted numbers, personal phone numbers including home and cellular phones. When service is provided by carriers (redirect to a PSTN leg, or a SIP URI provided by the service provider) there may be tolls associated with the forwarded leg. Rather than penalizing the UAC, it would be preferable if the toll for the forwarded leg be attributed to the forwarding server.

A final example to round up the motivation for this response code – SIP based call centers are becoming popular. These call centers may be geographically distributed to ensure 24/7 coverage and a single number or SIP URI is published. Redirect servers may employ a variety of techniques (including time-of-day routing) to send the call to the right call center – here too it is a requirement to perform recursion locally without having to send the redirection to the client. To remedy this situation, this document proposes a new response code, 303 Proxy Redirect.

2. Server Behavior

This section looks at the server that generates the new 303 Proxy Redirect response code. The INVITE from the originating client may land at either the UAS that represents the Request-URI or a redirect server within the domain responsible for the Request URI. As per (1) there is no real restriction on the URI format chosen – it could be tel:, fax:, mailto: etc

2.1. User Agent Servers

When an INVITE reaches the UAS (which may be a client representing the callee), the server may redirect the call to URI(s) of its choice. This document makes no recommendation on how this feature works from a User Interface perspective, and how the decision to recur locally is arrived at. The UAS adds one or more URIs to the
Contact header of the 303 Proxy Redirect response and sends it back to the previous hop.

2.2. Redirection Servers

Redirection Servers are used to send back alternate locations by including one or more URIs in the Contact header field of a 3xx message. In this case the Redirection Server determines that this call is to be sent to an alternate destination that needs to be recursed within its domain. How the Redirect Server determines the need for local recursion is outside the scope of this document.

The Redirect Server adds one or more URI(s) to the Contact header of the 303 Proxy Redirect response and sends it back to the previous hop.

Question: RFC3261 indicates that the Redirect Server may actually perform the recursion - in which case the behavior will be that of the Proxy and is detailed in the next section.

3. Proxy Behavior

The Proxy receiving a 303 Proxy Redirect will typically be required to recurse on the URI(s) provided in the Contact header. However given that Proxies are already deployed there arise several alternatives:

- The receiving Proxy does not support 303 Proxy Redirect: in this case it should treat the 303 as a generic 3xx message and follow behavior outlined in (1).

- The receiving Proxy supports 303 Proxy Redirect but is not able to perform recursion for some reason - must send a 404 Not Found back towards Originating client

- The receiving Proxy supports 303 Proxy Redirect and is able to perform recursion. The proxy incorporates the URIs from the Contact header in the 303 Proxy Redirect into a target set. The target set may include any other URIs that the proxy derives (example by contacting a location service) and generates a new request towards the URI(s) in the target set.

it may happen such that the Proxy in the domain responsible for the Request URI does not support 303, while the Proxy in the callers domain does. If the Proxy in the callers domain is stateless it may attempt to recurse on the URIs received in the 303 Proxy Redirect. Stateful Proxies may be able to avoid recursion, however the behavior
is not guaranteed.

Question: is it necessary to find a way to block this behavior i.e. find a way for a Proxy to figure that it is not responsible for the callee's domain? A check of the To: header may not be sufficient and a new mechanism may be required.

3.1. Example Flow

This call flow follows the classic SIP Trapezoid as detailed in [1]

<table>
<thead>
<tr>
<th>ALICE's Softphone</th>
<th>atlanta.com</th>
<th>biloxy.com proxy</th>
<th>Bob's phone</th>
<th>CAROL's phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>F1 INVITE</td>
<td></td>
<td>F2 INVITE</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>F3 INVITE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>F4 303 Proxy Redirect</td>
<td></td>
<td>F5 181 Call Being Forwarded</td>
<td></td>
</tr>
<tr>
<td></td>
<td>F6 INVITE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>F7 200 OK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>F8 ACK</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Note that Bob's phone generates F4 303 Proxy Redirect. Bob's proxy at biloxy.com recurses on the Contact header within the 303 Proxy Redirect and sends the F6 INVITE to Carol's phone. The F5 181 (Call is Being Forwarded) is optional.

4. Client Behavior

Though the 303 Proxy Redirect is expected to be handled by a Proxy in
the domain of the callee, there are circumstances under which the 303 may make it all the way back to the originating client. An example of such a situation is when intermediate Proxy does not support the 303 Proxy Redirect behavior. Upon receiving the 303 Proxy Redirect, the client may include the URIs received in the Contact Header into a target set and recurse upon it.

As routing to some of these URIs may fail (example private/internally routable URIs) or an additional toll may be associated with this new session, it may be useful if some sort of warning or alert be provided to the user before proceeding to send a new request to addresses derived from the 303.

5. 303 Proxy Redirect

303 Proxy Redirect response code allows the client or redirect server the ability cause a proxy to retarget an INVITE, rather than send the INVITE all the way back to the calling UAC. Proxys that do not support 303 are expected to treat it as a generic 3xx as per behavior defined in (1).

6. IANA Considerations

This section registers a new SIP response code according to the procedures of RFC 3261. RFC Number: RFC XXXX [[NOTE TO IANA: Please replace XXXX with the RFC number of this specification]] Response Code Number: 303 Default Reason Phrase: Proxy Redirect

7. Security Considerations

TBD.

8. Changes from 00

The changes from the 00 version of the text:

- Added more verbiage around the motivation for this draft
- Clarified behavior for SIP entities that produce and consume the 303 Proxy Redirect
- Added call flows
9. Acknowledgements

Thanks to the SIPPING Mailing list (specifically Jonathan R., and Keith Drage who provided comments).

10. Normative References


Authors’ Addresses

Rajesh Ramanathan
Microsoft Corporation
One Microsoft Way
Redmond, WA  98052
USA

Email: rajeshra@microsoft.com

Sriram Parameswar
Microsoft Corporation
One Microsoft Way
Redmond, WA  98052
USA

Email: srirampa@microsoft.com

Mohammad Vakil
Microsoft Corporation
One Microsoft Way
Redmond, WA  98052
USA

Email: mvakil@microsoft.com
Full Copyright Statement

Copyright (C) The IETF Trust (2007).

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.

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, THE IETF TRUST 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.

Intellectual Property

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.

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).