              Suppression of Session Initiation Protocol (SIP)
                     REFER Method Implicit Subscription

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.

Copyright Notice

Abstract

   The Session Initiation Protocol (SIP) REFER extension as defined in
   RFC 3515 automatically establishes a typically short-lived event
   subscription used to notify the party sending a REFER request about
   the receiver's status in executing the transaction requested by the
   REFER. These notifications are not needed in all cases. This
   specification provides a way to prevent the automatic establishment
   of an event subscription and subsequent notifications using a new SIP
   extension header field that may be included in a REFER request.

Table of Contents

1. Introduction

   The REFER specification [3] specifies that every REFER creates an
   implicit subscription between the REFER-Issuer and the REFER-
   Recipient.

   This document defines a new SIP header field: "Refer-Sub" meaningful
   within a REFER transaction only. This header field, when set to
   "false", specifies that a REFER-Issuer requests that the REFER-
   Recipient doesn't establish an implicit subscription and the
   resultant dialog.

   This document defines a new option tag: "norefersub". This tag, when
   included in the Supported header field, indicates that a User Agent
   (UA) is capable of accepting a REFER request without creating an
   implicit subscription when acting as a REFER-Recipient.

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 RFC 2119 [1].

   To simplify discussions of the REFER method and its extensions, the
   three terms below are being used throughout the document:

   o REFER-Issuer: the UA issuing the REFER request

   o REFER-Recipient: the UA receiving the REFER request

   o REFER-Target: the UA designated in the Refer-To Uniform Resource
      Identifier (URI)

3. Motivation

   The REFER specification mandates that every REFER creates an implicit
   subscription between the REFER-Issuer and the REFER-Recipient. This
   subscription results in at least one NOTIFY being sent from the
   REFER-Recipient to the REFER-Issuer. The REFER-Recipient may choose
   to cancel the implicit subscription with this NOTIFY. The REFER-
   Issuer may choose to cancel this implicit subscription with an
   explicit SUBSCRIBE (Expires: 0) after receipt of the initial NOTIFY.

   One purpose of requiring the implicit subscription and initial NOTIFY
   is to allow for the situation where the REFER request gets forked and
   the REFER-Issuer needs a way to see the multiple dialogs that may be
   established as a result of the forked REFER. This is the same
   approach used to handle forking of SUBSCRIBE [4] requests. Where the

   REFER-Issuer explicitly specifies that forking not occur, the
   requirement that an implicit subscription be established is
   unnecessary.

   Another purpose of the NOTIFY is to inform the REFER-Issuer of the
   progress of the SIP transaction that results from the REFER at the
   REFER-Recipient. In the case where the REFER-Issuer is already aware
   of the progress of the requested operation, such as when the REFER-
   Issuer has an explicit subscription to the dialog event package at
   the REFER-Recipient, the implicit subscription and resultant NOTIFY
   traffic related to the REFER can create an unnecessary network
   overhead.

4. Definitions

   This document defines a new SIP header field: "Refer-Sub". This
   header field is meaningful and MAY be used with a REFER request and
   the corresponding 2XX response only. This header field set to
   "false" specifies that a REFER-Issuer requests that the REFER-
   Recipient doesn't establish an implicit subscription and the
   resultant dialog. Note that when using this extension, the REFER
   remains a target refresh request (as in the default case -- when the
   extension is not used).

   This document adds the following entry to Table 2 of [2]. The
   additions to this table are also provided for extension methods at
   the time of publication of this document. This is provided as a
   courtesy to the reader and is not normative in any way:

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	MSG
Refer-Sub	R, 2xx		-	-	-	-	-	-	-

Header field	where	SUB	NOT	REF	INF	UPD	PRA	PUB
Refer-Sub	R, 2xx	-	-	o	-	-	-	-

   The Refer-Sub header field MAY be encrypted as part of end-to-end
   encryption.

   The syntax of the header field follows the BNF defined below:

    Refer-Sub   = "Refer-Sub" HCOLON refer-sub-value *(SEMI exten)
    refer-sub-value = "true" / "false"
    exten   = generic-param

   where the syntax of generic-param is defined in [2].

   The "Refer-Sub" header field set to "false" MAY be used by the REFER-
   Issuer only when the REFER-Issuer can be certain that the REFER
   request will not be forked.

   If the REFER-Recipient supports the extension and is willing to
   process the REFER transaction without establishing an implicit
   subscription, it MUST insert the "Refer-Sub" header field set to
   "false" in the 2xx response to the REFER-Issuer. In this case, no
   implicit subscription is created. Consequently, no new dialog is
   created if this REFER was issued outside any existing dialog.

   If the REFER-Issuer inserts the "Refer-Sub" header field set to
   "false", but the REFER-Recipient doesn't grant the suggestion (i.e.,
   either does not include the "Refer-Sub" header field or includes the
   "Refer-Sub" header field set to "true" in the 2xx response), an
   implicit subscription is created as in the default case.

   This document also defines a new option tag, "norefersub". This tag,
   when included in the Supported header field, specifies that a User
   Agent (UA) is capable of accepting a REFER request without creating
   an implicit subscription when acting as a REFER-Recipient.

   The REFER-Issuer can know the capabilities of the REFER-Recipient
   from the presence of the option tags in the Supported header field of
   the dialog initiating request or response. Another way of learning
   the capabilities would be by using presence, such as defined in [6].
   However, if the capabilities of the REFER-Recipient are not known,
   using the "norefersub" tag with the Require header field is NOT
   RECOMMENDED. This is due to the fact that in the event the REFER-
   Recipient doesn't support the extension, in order to fall back to the
   normal REFER, the REFER-Issuer will need to issue a new REFER
   transaction thus resulting in additional round-trips.

   As described in Section 8.2.2.3 in [2], a REFER-Recipient will reject
   a REFER request containing a Require: norefersub header field with a
   420 (Bad Extension) response unless it supports this extension. Note
   that Require: norefersub can be present with a Refer-Sub: false
   header field.

5. Preventing Forking of REFER Requests

   The REFER specification allows for the possibility of forking a REFER
   request that is sent outside of an existing dialog. In addition, a
   proxy may fork an unknown method type. Should forking occur, the
   sender of the REFER with "Refer-Sub" will not be aware as only a
   single 2xx response will be forwarded by the forking proxy. As a
   result, the responsibility is on the issuer of the REFER with "Refer-
   Sub" to ensure that no forking will result.

If a REFER request to a given Request-URI might fork, the REFER-
Issuer SHOULD NOT include the "Refer-Sub" header field. The REFER-
Issuer SHOULD use standardized mechanisms for ensuring the REFER
request does not fork. In absence of any other mechanism, the
Request-URI of the REFER request SHOULD have Globally Routable User
Agent URI (GRUU) properties according to the definitions of [5] as
those properties ensure the request will not fork.

6. Example

   An example of REFER that suppresses the implicit subscription is
   shown below. Note that the conventions used in the SIP Torture Test
   Messages [7] document are reused, specifically the <allOneLine> tag.

   REFER sip:pc-b@example.com SIP/2.0
   Via: SIP/2.0/TCP issuer.example.com;branch=z9hG4bK-a-1
   From: <sip:a@example.com>;tag=1a
   <allOneLine>
   To: sip:b@example.com;opaque=urn:uuid:f8
   1d4fae-7dec-11d0-a765-00a0c91e6bf6;grid=99a
   </allOneLine>
   Call-ID: 1@issuer.example.com
   CSeq: 234234 REFER
   Max-Forwards: 70
   Refer-To: <sip:c@example.com;method=INVITE>
   Refer-Sub: false
   Supported: norefersub
   Contact: sip:a@issuer.example.com
   Content-Length: 0

7. IANA Considerations

   This document registers a new SIP header field "Refer-Sub". This
   header field is only meaningful for the REFER request defined in RFC
   3515 [3] and the corresponding response. The following information
   has been added to the SIP Header field sub-registry in the SIP
   Parameters Registry:

   o Header Name: Refer-Sub

   o Compact Form: None

   o Reference: RFC 4488

   This document also registers a new SIP option tag, "norefersub",
   adding it to the SIP Option Tags sub-registry in the SIP Parameters
   Registry. The required information for this registration, as
   specified in RFC 3261 [2], is:

o Name: norefersub

o Description: This option tag specifies a User Agent ability of
   accepting a REFER request without establishing an implicit
   subscription (compared to the default case defined in RFC 3515
   [3]).

8. Security Considerations

   The purpose of this SIP extension is to modify the expected behavior
   of the REFER-Recipient. The change in behavior is for the REFER-
   Recipient not to establish a dialog and not to send NOTIFY messages
   back to the REFER-Issuer. As such, a malicious inclusion of a
   "Refer-Sub" header field set to "false" reduces the processing and
   state requirements on the recipient. As a result, its use in a
   denial-of-service attack seems limited.

   On the other hand, by inserting a "Refer-Sub" header field set to
   "false", a man-in-the-middle (MitM) can potentially exploit the
   mechanism for easier (than an interception) suppression of the
   notifications from the REFER-Receiver without the REFER-Issuer
   noticing it. Also, by removing a "Refer-Sub" header field set to
   "false", a MitM can cause the REFER-Receiver to generate
   notifications over the implicit dialog that otherwise had been
   suppressed by the REFER-Issuer.

   To protect against these kinds of MitM attacks, integrity protection
   should be used. For example, the REFER-Issuer could use S/MIME as
   discussed in RFC 3261 [2] to protect against these kinds of attacks.

9. Acknowledgements

   The SIP community would like to thank Sriram Parameswar for his
   ideas, originally presented in "Suppressing Refer Implicit
   Subscription" (October 2002), which served as the basis for this
   specification.

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

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

10.2. Informative References

   [5] Rosenberg, J., "Obtaining and Using Globally Routable User Agent
        (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)",
         Work in Progress, October 2005.

   [6] Lonnfors, M. and K. Kiss, "Session Initiation Protocol (SIP)
        User Agent Capability Extension to Presence Information Data
        Format (PIDF)", Work in Progress, January 2006.

   [7] Sparks, R., Ed., Hawrylyshen, A., Johnston, A., Rosenberg, J.,
        and H. Schulzrinne, "Session Initiation Protocol (SIP) Torture
        Test Messages", RFC 4475, May 2006.

Author's Address

   Orit Levin
   Microsoft Corporation
   One Microsoft Way
   Redmond, WA 98052
   USA

   Phone: 425-722-2225
   EMail: oritl@microsoft.com

Full Copyright Statement

Intellectual Property

Acknowledgement