Sieve Notification Mechanism: mailto
draft-ietf-sieve-notify-mailto-06

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 28, 2008.

Copyright Notice

Copyright (C) The IETF Trust (2008).
Abstract

This document describes a profile of the Sieve extension for notifications, to allow notifications to be sent by electronic mail.

Table of Contents

1. Introduction ...................................................... 3
1.1. Overview .......................................................... 3
1.2. Conventions used in this document .............................. 3

2. Definition ............................................................ 4
2.1. Notify parameter "method" ....................................... 4
2.2. Test notify_method_capability .................................. 4
2.3. Notify tag ":from" ............................................... 4
2.4. Notify tag ":importance" ........................................ 4
2.5. Notify tag ":options" ............................................ 4
2.6. Notify tag ":message" ............................................ 4
2.7. Other Definitions ................................................ 5

3. Examples .............................................................. 8

4. Internationalization Considerations .............................. 10

5. Security Considerations ........................................... 11

6. IANA Considerations ................................................ 12
6.1. Registration of notification mechanism ........................ 12
6.2. New registry for Auto-Submitted header field keywords . 12
6.3. Initial registration of Auto-Submitted header field keywords ........................................... 12

7. References ........................................................... 14
7.1. Normative References ............................................ 14
7.2. Non-Normative References ...................................... 14

Authors’ Addresses .................................................... 15

Intellectual Property and Copyright Statements ..................... 16
1. Introduction

1.1. Overview

The [Notify] extension to the [Sieve] mail filtering language is a framework for providing notifications by employing URIs to specify the notification mechanism. This document defines how [mailto] URIs are used to generate notifications by e-mail.

1.2. Conventions used in this document

Conventions for notations are as in [Sieve] section 1.1, including the use of [Kwds].

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 [Kwds].
2. Definition

The mailto mechanism results in the sending of a new email message (a "notification message") to notify a recipient about a "triggering message".

2.1. Notify parameter "method"

The mailto notification mechanism uses standard mailto URIs as specified in [mailto].

2.2. Test notify_method_capability

The notify_method_capability test for "online" may return "yes" or "no" only if the Sieve processor can determine with certainty whether or not the recipients of the notification message are online and logged in. Otherwise, the test returns "maybe" for this notification method.

2.3. Notify tag ":from"

The :from tag overrides the default sender of the notification message. "Sender", here, refers to the value used in the [RFC2822] "From" header. Implementations MAY also use this value in the [RFC2821] "MAIL FROM" command (the "envelope sender"), or they may prefer to establish a mailbox that receives bounces from notification messages.

2.4. Notify tag ":importance"

The :importance tag has no special meaning for this notification mechanism, and this specification puts no restriction on its use. Implementations MAY use the value of :importance to set a priority or importance indication on the notification message (perhaps a visual indication, or perhaps making use of one of the non-standard but commonly used message headers).

2.5. Notify tag ":options"

This tag is not used by the mailto method.

2.6. Notify tag ":message"

The value of this tag, if it is present, is used as the subject of the notification message, and overrides all other mechanisms for determining the subject (as described below). Its value SHOULD NOT normally be truncated, though it may be sensible to truncate an excessively long value.
2.7. Other Definitions

Because the receipt of an email message is generating another email message, implementations MUST take steps to avoid mail loops. The notification message contains the "Received:" fields from the triggering message to allow loop detection as described in [RFC2821], section 6.2. The REQUIRED inclusion of an "Auto-Submitted:" field, as described in the message composition guidelines, will also help in loop detection and avoidance.

Implementations MUST NOT trigger notifications for messages containing "Auto-Submitted:" header fields with any value other than "No".

Implementations MUST allow messages with empty envelope senders to trigger notifications.

Because this notification method uses a store-and-forward system for delivery of the notification message, the Sieve processor should not have a need to retry notifications. Therefore, implementations of this method SHOULD use normal mechanisms for submitting SMTP messages and for retrying the initial submission. Once the notification message is submitted, implementations MUST NOT resubmit it, as this is likely to result in multiple notifications, and increases the danger of message loops.

The overall notification message is composed using the following guidelines (see [RFC2822] for references to message header fields):

- If the envelope sender of the triggering message is empty, the envelope sender of the notification message MUST be empty as well, to avoid message loops. Otherwise, the envelope sender of the notification message is set to the value of the ":from" parameter to the notify action, if one is specified, has email address syntax and is valid according to the implementation specific security checks (see Section 3.3 of [Notify]). If ":from" is not specified or is not valid, the envelope sender of the notification message is set either to the envelope "to" field from the triggering message, as used by Sieve, or to a fixed email address (so it "comes from the notification system"), at the discretion of the implementation. This may not be overridden by a "from" URI header, and any such URI header MUST be ignored.

- The header field "Auto-Submitted: sieve-notify" MUST be included in the notification message (see [RFC3834]). This is to reduce the likelihood of message loops, by tagging this as an automatically generated message. Among other results, it will cause the notification message not to generate further
notifications. mailto URI headers with hname "auto-submitted" are considered unsafe and MUST be ignored.

- The "From:" header field of the notification message is set to the value of the ":from" parameter to the notify action, if one is specified, has email address syntax and is valid according to the implementation specific security checks (see Section 3.3 of Notify). If ":from" is not specified or is not valid, the "From:" header field of the notification message is set either to the envelope "to" field from the triggering message, as used by Sieve, or to a fixed email address (so it "comes from the notification system"), at the discretion of the implementation. This may not be overridden by a "from" URI header, and any such URI header MUST be ignored.

- The "To:" header field and the envelope recipient(s) of the notification message are set to the address(es) specified in the URI (including any URI headers where the hname is "to").

- The "Subject:" field of the notification message contains the value defined by the :message notify tag, as described in Notify. If there is no :message tag and there is a "subject" header on the URI, then that value is used. If that is also absent, the subject is retained from the triggering message. Note that Sieve [Variables] can be used to advantage here, as shown in the example in Section 3.

- If the mailto URI contains a "body" header, the value of that header is used as the body of the notification message. If there is no "body" header, it is up to the implementation whether to leave the body empty or to use an excerpt of the original message.

- The "Received:" fields from the triggering message are retained in the notification message, as these may help detect and prevent mail loops. URI headers with hname "received" are considered unsafe, and will be ignored.

- Other header fields of the notification message that are normally related to an individual new message (such as "Message-ID" and "Date") are generated for the notification message in the normal manner. Any URI headers with those names are ignored. Further, the "Date" header serves as the notification timestamp defined in Notify.

- All other header fields of the notification message either are as specified by URI headers, or have implementation-specific values; their values are not defined here. It is suggested that the implementation capitalizes the first letter of URI headers and
adds a space character after the colon between the mail header name and value when adding URI headers to the message.
3. Examples

Triggering message (received by recipient@example.org):

Return-Path: <knitting-bounces@example.com>
Received: from mail.example.com by mail.example.org
    for <recipient@example.org>; Wed, 7 Dec 2005 05:08:02 -0500
Received: from hobbies.example.com by mail.example.com
    for <knitting@example.com>; Wed, 7 Dec 2005 02:00:26 -0800
Message-ID: <1234567.89ABCDEF@example.com>
Date: Wed, 07 Dec 2005 10:59:19 +0100
Precedence: list
List-Id: Knitting Mailing List <knitting@example.com>
Sender: knitting-bounces@example.com
Errors-To: knitting-bounces@example.com
From: "Jeff Smith" <jeff@hobbies.example.com>
To: "Knitting Mailing List" <knitting@example.com>
Subject: [Knitting] A new sweater

I just finished a great new sweater!

Sieve script (run on behalf of recipient@example.org):

require ["notify", "variables"];

if header :contains "list-id" "knitting.example.com" {
    if header :matches "Subject" "[*] *" {
        notify :message "From ${1} list: ${2}
            :importance "3"
            "mailto:0123456789@sms.example.net";
    }
}

Notification message:

Received: from mail.example.com by mail.example.org
    for <recipient@example.org>; Wed, 7 Dec 2005 05:08:02 -0500
Received: from hobbies.example.com by mail.example.com
    for <knitting@example.com>; Wed, 7 Dec 2005 02:00:26 -0800
Date: Wed, 7 Dec 2005 05:08:55 -0500
Message-ID: <A2299BB.FF7788@example.org>
Auto-Submitted: sieve-notify
From: <recipient@example.org>
To: <0123456789@sms.example.net>
Subject: From Knitting list: A new sweater
Note that:

- Fields such as "Message-ID:" and "Date:" were generated afresh for the notification message, and do not relate to the triggering message.

- Additional "Received:" fields will be added to the notification message in transit; the ones shown were copied from the triggering message.

- If this message should appear at the mail.example.org server again, the server can use the presence of a "mail.example.org" received line to recognize that. The Auto-Submitted header field is also present to tell the server to avoid sending another notification.
4. Internationalization Considerations

This specification introduces no specific internationalization issues that are not already addressed in [Sieve] and in [Notify].
5. Security Considerations

Sending a notification is comparable with forwarding mail to the notification recipient. Care must be taken when forwarding mail automatically, to ensure that confidential information is not sent into an insecure environment.

The automated sending of email messages exposes the system to mail loops, which can cause operational problems. Implementations of this specification MUST protect themselves against mail loops (see Section 2.7).

Additional security considerations are discussed in [Sieve] and in [Notify].
6. IANA Considerations

6.1. Registration of notification mechanism

The following template specifies the IANA registration of the Sieve notification mechanism specified in this document:

To: iana@iana.org
Subject: Registration of new Sieve notification mechanism
Mechanism name: mailto
Mechanism URI: RFC2368
Mechanism-specific tags: none
Standards Track/IESG-approved experimental RFC number: this RFC
Person and email address to contact for further information:
   Michael Haardt <michael.haardt@freenet.ag>

This information should be added to the list of sieve notification mechanisms given on http://www.iana.org/assignments/sieve-notification.

6.2. New registry for Auto-Submitted header field keywords

Because [RFC3834] does not define a registry for new keywords used in the Auto-Submitted header field, we define one here, to be created as http://www.iana.org/assignments/auto-submitted-keywords. This defines the template to be used to register new keywords.

To: iana@iana.org
Subject: Registration of new auto-submitted header field keyword
Keyword value: [the text value of the field]
Description: [a brief explanation of the purpose of this value]
Standards Track/IESG-approved experimental RFC number: [identifies the specification that defines the value being registered]
Contact: [name and email address to contact for further information]

6.3. Initial registration of Auto-Submitted header field keywords

The following are the initial keywords to be registered for the Auto-Submitted header field, to be entered in http://www.iana.org/assignments/auto-submitted-keywords.

Keyword value: no
Description: Indicates that a message was NOT automatically generated, but was created by a human. It is the equivalent to the absence of an Auto-Submitted header altogether.
Standards Track/IESG-approved experimental RFC number: RFC3834
Contact: Keith Moore <moore@cs.utk.edu>
Keyword value: auto-generated
Description: Indicates that a message was generated by an automatic
process, and is not a direct response to another message.
Standards Track/IESG-approved experimental RFC number: RFC3834
Contact: Keith Moore <moore@cs.utk.edu>

Keyword value: auto-replied
Description: Indicates that a message was automatically generated as
a direct response to another message.
Standards Track/IESG-approved experimental RFC number: RFC3834
Contact: Keith Moore <moore@cs.utk.edu>

Keyword value: sieve-notify
Description: Indicates that a message was generated by a Sieve
notification system.
Standards Track/IESG-approved experimental RFC number: this RFC
Contact: Michael Haardt <michael.haardt@freenet.ag>
7. References

7.1. Normative References


7.2. Non-Normative References


Authors’ Addresses

Barry Leiba
IBM T.J. Watson Research Center
19 Skyline Drive
Hawthorne, NY  10532
US

Phone: +1 914 784 7941
Email: leiba@watson.ibm.com

Michael Haardt
freenet AG
Willstaetter Str. 13
Duesseldorf, NRW  40549
Germany

Phone: +49 241 53087 520
Email: michael.haardt@freenet.ag