Sieve Working Group
Internet-Draft
Updates: 5228 (if approved)
Intended status: Standards Track
Expires: March 14, 2013

B. Leiba
Huawei Technologies
September 10, 2012

Support for Internet Message Access Protocol (IMAP) Events in Sieve
draft-ietf-sieve-imap-sieve-08

Abstract

Sieve defines an email filtering language that can, in principle, plug into any point in the processing of an email message. As defined in the base specification, it plugs into mail delivery. This document defines how Sieve can plug into points in the IMAP protocol where messages are created or changed, adding the option of user-defined or installation-defined filtering (or, with Sieve extensions, features such as notifications). Because this requires future Sieve extensions to specify their interactions with this one, this document updates the base Sieve specification, RFC 5228.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on March 14, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 4
1.1. Overview 4
1.2. Differences Between IMAP Events and Mail Delivery 4
1.3. Conventions used in this document 5

2. The IMAP Events in Sieve Extension 6
2.1. The "imapsieve" Capability Strings 6
2.2. Existing IMAP Functions Affected by IMAP events in Sieve 6
2.2.1. The IMAP APPEND Command 7
2.2.2. The IMAP MULTIAPPEND Command 7
2.2.3. The IMAP COPY Command 7
2.2.4. Changes to IMAP Message Flags 7
2.3. New Functions Defined by IMAP events in Sieve 8
2.3.1. Interaction with Metadata 8

3. Applicable Sieve Actions and Interactions 10
3.1. The Implicit Keep 10
3.2. The Keep Action 10
3.3. The Fileinto Action 10
3.4. The Redirect Action 11
3.5. The Discard Action 11
3.6. The Notify Action 12
3.7. The Addheader and Deleteheader Actions 12
3.8. The Setflag, Deleteflag, and Removeflag Actions 12
3.9. MIME Part Tests and Replacement 12
3.10. Spamtest and Virustest 13
3.11. Inapplicable Actions 13
3.12. Future Sieve Actions 13

4. Interaction With Sieve Environment 14
4.1. Base Sieve Environment Items: location and phase 14
4.2. New Sieve Environment Items: imapuser and imapemail . . . 14
4.3. New Sieve Environment Item: cause 14
4.4. New Sieve Environment Item: mailbox 15
4.5. New Sieve Environment Item: changedflags 15
4.6. Interaction With Sieve Tests (Comparisons) 15

5. Examples 16

6. Security Considerations 17

7. IANA Considerations 18
7.1. Registration of "imapsieve" IMAP capability 18
7.2. Registration of "imapsieve" Sieve extension 18
7.3. Registration of Sieve Environment Items 18
7.3.1. Registration of Sieve Environment Item: cause 18
7.3.2. Registration of Sieve Environment Item: mailbox 19
7.3.3. Registration of Sieve Environment Item: changedflags . . . 19
7.3.4. Registration of Sieve Environment Item: imapuser 19
7.3.5. Registration of Sieve Environment Item: imapemail 19
7.4. Registration of IMAP METADATA Mailbox Entry Name 19
7.5. Registration of IMAP METADATA Server Entry Name 20

8. References 21
8.1. Normative References 21
8.2. Informative References 21

Author's Address 23

1. Introduction

1.1. Overview

Some applications have a need to apply Sieve filters [RFC5228] in contexts other than initial mail delivery. This is especially true in diverse service environments, such as when the client is sporadically connected, is connected through a high-latency or high-cost channel, or is on a limited-function device. For such clients, it may be very important, for higher performance and reliability, to take advantage of server capabilities, including those provided by Sieve filtering (and Sieve extensions, such as Notify [RFC5435]).

This specification defines extensions to IMAP [RFC3501] to support the invocation of Sieve scripts at times when the IMAP server creates new messages or modifies existing ones. It also defines how Sieve scripts will process these invocations. Support for IMAP events in Sieve requires support for IMAP Metadata [RFC5464] and Sieve Environment [RFC5183] as well, because Metadata is used to associate scripts with IMAP mailboxes and Environment defines an important way for Sieve scripts to test the conditions under which they have been invoked.

Because this requires future Sieve extensions to specify their interactions with this one (see Section 3.12), this document updates the base Sieve specification, RFC 5228.

1.2. Differences Between IMAP Events and Mail Delivery

Invoking Sieve scripts in a context other than initial mail delivery introduces new situations, which changes the applicability of Sieve features and creates implementation challenges and user interface issues. This section discusses some of those differences, challenges, and issues.

At times other than message delivery, delivery "envelope" information might not be available. With messages added through IMAP APPEND, there might be no way to even guess who the intended recipient is, and no concept of who "sent" the message. Sieve actions that relate to contacting the sender, for example, will not be applicable.

Because IMAP events will often be triggered by user actions, and because user interfaces allow bulk actions that differ from individual message arrival, it now becomes possible for a single user action, such as drag-and-drop, to initiate Sieve script processing on a large number of messages at once. Implementations will have to deal with such situations as a "COPY" action or flag changes on dozens, or even thousands of messages.

Other issues might surface as this extension is deployed and experience with it develops.

1.3. Conventions used in this document

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 [RFC2119].

2. The IMAP Events in Sieve Extension

2.1. The "imapsieve" Capability Strings

An IMAP server advertises support for IMAP events in Sieve through the "imapsieve" capability. A server that advertises "imapsieve" is claiming to be in compliance with this specification in all aspects. The syntax of the "imapsieve" capability string is defined as follows:

```
capability /=  "IMAPSIEVE=" sieveurl-server
        ; <sieveurl-server> is defined in RFC 5804, Section 3
```

Only one "imapsieve" capability string, specifying one sieveurl-server, is allowed to be present. The sieveurl-server identifies the ManageSieve server that clients need to contact for managing Sieve scripts associated with this IMAP server.

The corresponding Sieve implementation uses the Sieve capability string "imapsieve", and Sieve scripts that depend upon the IMAP events MUST include that string in their "required" lists.

Implementations that support IMAP events in Sieve MUST also support IMAP Metadata [RFC5464] and Sieve Environment [RFC5183], because Metadata is used to associate scripts with IMAP mailboxes and Environment defines an important way for Sieve scripts to test the conditions under which they have been invoked. Notwithstanding the support requirement, scripts that directly use Environment MUST also include its capability string in their "required" lists.

2.2. Existing IMAP Functions Affected by IMAP events in Sieve

The subsections below describe in detail the IMAP commands and situations on which IMAP events in Sieve have an effect. Not all Sieve actions make sense in the case of messages affected by IMAP commands. See Section 3 for details.

It's important to note that since the base Sieve specification (see [RFC5228]) and its extensions define functions for scripts that are invoked during initial mail delivery, those function definitions are necessarily tailored to and limited by that context. This document extends those function definitions for use during IMAP events. By nature of that, Sieve functions, in this extended context, may behave somewhat differently, though their extended behaviour will still be consistent with the functions' goals.

If more than one message is affected at the same time, each message triggers the execution of a Sieve script separately. The scripts MAY

be run in parallel.

2.2.1. The IMAP APPEND Command

A message may be added to a mailbox through the IMAP APPEND command. In a server that advertises "imapsieve", new messages added in this way MUST trigger the execution of a Sieve script, subject to the settings defined through Metadata (see Section 2.3.1).

2.2.2. The IMAP MULTIAPPEND Command

If the IMAP server supports the IMAP MultiAppend extension [RFC3502], messages may be added to a mailbox through the IMAP MULTIAPPEND command. In a server that advertises "imapsieve", new messages added in this way MUST trigger the execution of a Sieve script, as with the APPEND command, also subject to the settings defined through Metadata.

2.2.3. The IMAP COPY Command

One or more messages may be added to a mailbox through the IMAP COPY command. In a server that advertises "imapsieve", new messages added in this way MUST trigger the execution of a Sieve script, subject to the settings defined through Metadata.

2.2.4. Changes to IMAP Message Flags

One or more existing messages can have their flags changed in a number of ways, including:

- The FETCH command (may cause the \Seen flag to be set).

- The STORE command (may cause the \Answered, \Deleted, \Draft, \Flagged, and \Seen flags to be set or reset, and may cause keywords to be set or reset).

- The invocation of a Sieve script on an existing message, where the Sieve implementation supports the IMAP4Flags extension [RFC5232] and the script uses one of the actions defined in that extension.

In a server that advertises "imapsieve", messages whose flags are changed in any way (except as explained in the next sentence) MUST trigger the execution of a Sieve script, subject to the settings defined through Metadata. The exception is that in order to avoid script loops, flag changes that are made as a result of a script that was itself invoked because of flag changes SHOULD NOT result in a further invocation of the script. In any case, implementations MUST take steps to avoid such loops.

For flag-change events, the Sieve script will see the message flags as they are AFTER the changes.

2.3. New Functions Defined by IMAP events in Sieve

2.3.1. Interaction with Metadata

Support for IMAP events in Sieve requires support for IMAP Metadata [RFC5464] as well, since the latter is used to associate scripts with IMAP mailboxes.

When an applicable event occurs on an IMAP mailbox, if there is an IMAP metadata entry named "/shared/imapsieve/script" for the mailbox, that entry is used. If there is not, but there is an IMAP metadata entry named "/shared/imapsieve/script" for the server, that entry is used (providing a way to define a global script for all mailboxes on a server). If neither entry exists, then no script will be invoked.

If a "/shared/imapsieve/script" metadata entry was selected above, its value is used as the name of the Sieve script that will be invoked in response to the IMAP event. If the value is empty, then no script is run. The selection of which metadata entry to use happens before any examination of the contents of the entry. If the mailbox entry is selected and is then found to be unusable or empty, the server entry is not used as a backup: no script is run.

This specifies the mechanism for "activating" a script for a given mailbox (or for all mailboxes), but does not specify a mechanism for creating, storing, or validating the script. Implementations MUST support ManageSieve [RFC5804], and can use the PUTSCRIPT command to store the script without using the SETACTIVE command to activate it.

Script names used in "/shared/imapsieve/script" metadata entries are the script names used on the corresponding ManageSieve server. If a "/shared/imapsieve/script" metadata entry contains a script name that doesn't exist in the ManageSieve server, then no Sieve script will be invoked for IMAP Sieve events.

Only one Sieve script may currently be defined per mailbox, eliminating the complexity and possible ambiguity involved with coordinating the results of multiple scripts. Any sub-filtering is done in the Sieve script. For example, if it's only necessary to deal with flag changes, but not with new messages appended or copied, the Sieve script will still be invoked for all events, and the script is responsible for checking the event type.

The possibility is open for an extension to add support for multiple scripts -- for example, per-client scripts on a multi-client user's

inbox, or per-user scripts on a mailbox that is shared among users.

Because this metadata name is associated with the mailbox, there can (and it's expected that there will) be different scripts associated with events for different mailboxes. Indeed, most mailboxes will probably invoke no script at all.

3. Applicable Sieve Actions and Interactions

Since some Sieve actions relate specifically to the delivery of mail, not all actions and extensions make sense when the messages are created by other means or when changes are made to data associated with existing messages. This section describes how actions in the base Sieve specification, and those in extensions known at this writing, relate to this specification.

In addition to what is specified here, interactions noted in the individual specifications apply, and must be considered.

3.1. The Implicit Keep

For all cases that fall under IMAP events in Sieve, the implicit keep means that the message is treated as it would have been if no Sieve script were run. For APPEND, MULTIAPPEND and COPY, the message is stored into the target mailbox normally. For flag changes, the message is left in the mailbox. If actions have been taken that change the message, those changes are considered transient and MUST NOT be retained for any keep action (because IMAP messages are immutable). No error is generated, but the original message, without the changes, is kept.

3.2. The Keep Action

The keep action is applicable in all cases that fall under IMAP events in Sieve. Its behaviour is as described for implicit keep, in Section 3.1.

3.3. The Fileinto Action

If the Sieve implementation supports the fileinto action, that action is applicable in all cases that fall under IMAP events in Sieve. If the Copy extension [RFC3894] is available and the :copy option is specified, the implicit keep is retained; otherwise, fileinto cancels the implicit keep, as specified in the base Sieve specification.

For APPEND, MULTIAPPEND, and COPY, the message is stored into the fileinto mailbox IN ADDITION TO the original target mailbox. For flag changes, the message is COPIED into the fileinto mailbox, without removing the original. In all cases, fileinto always creates a new message, separate from the original.

If a keep action is not also in effect, the original message is then marked with the \Deleted flag (and a flag-change Sieve script is not invoked). The implementation MAY then expunge the original message (WITHOUT expunging other messages in the mailbox); alternatively, it

might choose to have expunges batched or done by a user. If the server does the expunge, the effect is as though a client had flagged the message and done a UID EXPUNGE (see [RFC4315]) on the affected message(s) only. Handling it this way allows clients to handle messages consistently, and avoids hidden changes that might invalidate their message caches.

3.4. The Redirect Action

The redirect action is applicable in all cases that fall under IMAP events in Sieve. It causes the message to be sent, as specified in the base Sieve specification, to the designated address. If the Copy extension [RFC3894] is available and the :copy option is specified, the implicit keep is retained; otherwise, redirect cancels the implicit keep, as specified in the base Sieve specification.

It's possible that a message processed in this way does not have the information necessary to be redirected properly. It might lack necessary header information, and there might not be appropriate information for the MAIL FROM command. In such cases, the "redirect" action uses Message Submission [RFC6409], and it is up to the Sieve engine to supply the missing information. The redirect address is, of course, used for the "RCPT TO", and the "MAIL FROM" SHOULD be set to the address of the owner of the mailbox. The message submission server is allowed, according to the Message Submission protocol, to perform necessary fix-up to the message (see Section 8 of RFC 6409). It can also reject the submission attempt, if the message is too ill-formed for submission.

For APPEND, MULTIAPPEND, and COPY, the message is stored into the target mailbox in addition to being redirected. For flag changes, the message remains in its original mailbox.

If a keep action is not also in effect, the original message is then marked with the \Deleted flag (and a flag-change Sieve script is not invoked). The implementation MAY then expunge the original message (WITHOUT expunging other messages in the mailbox); alternatively, it might choose to have expunges batched or done by a user. If the server does the expunge, the effect is as though a client had flagged the message and done a UID EXPUNGE (see [RFC4315]) on the affected message(s) only. Handling it this way allows clients to handle messages consistently, and avoids hidden changes that might invalidate their message caches.

3.5. The Discard Action

The discard action is applicable in all cases that fall under IMAP events in Sieve. For APPEND, MULTIAPPEND, and COPY, the message is

first stored into the target mailbox. If an explicit keep action is also in effect, the discard action now does nothing. Otherwise, the original message is then marked with the \Deleted flag (and a flag-change Sieve script is not invoked). The implementation MAY then expunge the original message (WITHOUT expunging other messages in the mailbox); alternatively, it might choose to have expunges batched or done by a user. If the server does the expunge, the effect is as though a client had flagged the message and done a UID EXPUNGE (see [RFC4315]) on the affected message(s) only. Handling it this way allows clients to handle messages consistently, and avoids hidden changes that might invalidate their message caches.

3.6. The Notify Action

If the Nofity extension [RFC5435] is available, the notify action is applicable in all cases that fall under IMAP events in Sieve. The result is that the requested notification is sent, and that the message is otherwise handled as it would normally have been.

3.7. The Addheader and Deleteheader Actions

If the EditHeader extension [RFC5293] is available, it can be used to make transient changes to header fields, which aren't saved in place, such as for "redirect" or "fileinto" actions. Because messages in IMAP mailboxes are immutable, such changes are not applicable for the "keep" action (explicit or implicit). See Section 3.1.

3.8. The Setflag, Deleteflag, and Removeflag Actions

Implementations of IMAP events in Sieve MUST also support the IMAP4Flags extension [RFC5232], and the actions associated with it are all applicable to any case that falls under IMAP events in Sieve.

It is worth noting also that the "hasflag" test that is defined in the IMAP4Flags extension might be particularly useful in scripts triggered by flag changes ("hasflag" will see the new, changed flags). The flag changes behave as though a client had made the change.

As explained above, in order to avoid script loops flag changes that are made as a result of a script that was itself invoked because of flag changes SHOULD NOT result in another script invocation. In any case, implementations MUST take steps to avoid such loops.

3.9. MIME Part Tests and Replacement

If the MIME Part Tests extension [RFC5703] is available, all of its functions can be used, but any changes made to the message, using the

"replace" or "enclose" action, MUST be considered transient, and are only applicable with actions such as "redirect" and "fileinto". Because messages in IMAP mailboxes are immutable, such changes are not applicable for the "keep" action (explicit or implicit). See Section 3.1.

3.10. Spamtest and Virustest

If the Spamtest and Virustest extensions [RFC5235] are available, they are applicable in all cases that fall under IMAP events in Sieve.

3.11. Inapplicable Actions

The following actions and extensions are not applicable to any case that falls under IMAP events in Sieve, because they are specifically designed to respond to delivery of a new email message. Their appearance in the "require" control or their use in an IMAP event MUST result in an error condition that will terminate the Sieve script:

reject [RFC5228]

ereject [RFC5429]

vacation [RFC5230]

Future extensions that are specifically designed to respond to delivery of a new email message will likewise not be applicable to this extension.

3.12. Future Sieve Actions

As noted above, future extensions that are specifically designed to respond to delivery of a new email message will not be applicable to this extension, because this extension does not involve acting at new-message delivery time.

In general, future extensions to Sieve that define new actions MUST specify the applicability of those actions to this specification.

4. Interaction With Sieve Environment

4.1. Base Sieve Environment Items: location and phase

The Sieve Environment extension defines a set of standard environment items (see [RFC5183], Section 4.1). Two of those items are affected when the script is invoked through an IMAP event.

The value of "location" is set to "MS" -- evaluation is being performed by a Message Store.

The value of "phase" is set to "post" -- processing is taking place after (or perhaps instead of, in the case of APPEND) final delivery.

4.2. New Sieve Environment Items: imapuser and imapemail

In the normal case, when Sieve is used in final delivery, there is no identity for the "filer" -- the user who is creating or changing the message. In this case, there is such an identity, and a Sieve script might want to access that identity.

Implementations MUST set and make available two new environment items:

"imapuser" -- the identity (login ID) of the IMAP user that caused the action. This MUST be the empty string if it is accessed during normal (final delivery) Sieve processing.

"imapemail" -- the primary email address of the IMAP user that caused the action (the user identified by "imapuser"). In some implementations, "imapuser" and "imapemail" might have the same value. This MUST be the empty string if it is accessed during normal (final delivery) Sieve processing.

4.3. New Sieve Environment Item: cause

Each mailbox uses a single script for all the change conditions described in this document (append, copy, flag changes). To support that, the implementation MUST set the Environment [RFC5183] item "cause", which contains the name of the action that caused the script to be invoked. Its value is one of the following:

- o APPEND (for invocations resulting from APPEND or MULTIAPPEND)

- o COPY (for invocations resulting from COPY)

- o FLAG (for invocations resulting from flag changes)

Future extensions might define new events and, thus, new causes. Such extensions will come with their own capability strings, and the events they define will only be presented when their capabilities are requested. Scripts that do not request those capabilities will not see those events, and will not encounter the new cause strings.

4.4. New Sieve Environment Item: mailbox

The implementation MUST set the Environment [RFC5183] item "mailbox" to the name of the mailbox that the affected message is in, in the case of existing messages, or is targeted to be stored into, in the case of new messages. The value of this item is fixed when the script begins, and, in particular, MUST NOT change as a result of any action, such as "fileinto".

4.5. New Sieve Environment Item: changedflags

If the IMAP4Flags extension [RFC5232] is available, AND the script was invoked because of flag changes to an existing message, the implementation MUST set the Environment [RFC5183] item "changedflags" to the name(s) of the flag(s) that have changed. If the script was not invoked because of flag changes, the value of this item MUST be the empty string. The script will not know from this item whether the flags have been set or reset, but it can use the "hasflag" test to determine the current value. See example 2 in Section 5 for an example of how this might be used.

4.6. Interaction With Sieve Tests (Comparisons)

Any tests against message envelope information, including the "envelope" test in the Sieve base specification, as well as any such test defined in extensions, are either inapplicable or have serious interoperability issues when performed at other than final-delivery time. Therefore, envelope tests MUST NOT be permitted in the cases described here, and their use MUST generate a runtime error.

This extension does not affect the operation of other tests or comparisons in the Sieve base specification.

5. Examples

Example 1:
If a new message is added to the "ActionItems" mailbox, a copy is sent to the address "actionitems@example.com".

```
require ["copy", "environment", "imapsieve"];

if anyof (environment :is "cause" "APPEND",
          environment :is "cause" "COPY")  {
    if environment :is "mailbox" "ActionItems" {
        redirect :copy "actionitems@example.com";
    }
}
```

Example 2:
If the script is called for any message with the \Flagged flag set (tested through the IMAP4Flags extension [RFC5232]), a notification is sent using the Notify extension [RFC5435]. No notification will be sent, though, if we're called with an existing message that already had that flag set.

```
require ["enotify", "imap4flags", "variables",
         "environment", "imapsieve"];

if environment :matches "mailbox" "*" {
    set "mailbox" "${1}";
}

if allof (hasflag "\\Flagged",
          not environment :contains "changedflags" "\\Flagged") {
  notify :message "Important message in ${mailbox}"
      "xmpp:tim@example.com?message;subject=SIEVE";
}
```

Example 3:
This shows an example IMAP CAPABILITY response when this extension is supported. The client has done STARTTLS with the server, and is now inspecting capabilities. (The untagged CAPABILITY response is split here for readability only, but will be in one response message.)

```
C:  A01 CAPABILITY
S:  * CAPABILITY IMAP4rev1 AUTH=PLAIN UIDPLUS LIST-EXTENDED
      ACL IMAPSIEVE=sieve://sieve.example.com MULTISEARCH
S:  A01 OK done
```

6. Security Considerations

It is possible to introduce script processing loops by having a Sieve script that is triggered by flag changes use the actions defined in the IMAP4Flags extension [RFC5232]. Implementations MUST take steps to prevent such loops. One way to avoid this problem is that if a script is invoked by flag changes, and that script further changes the flags, those flag changes SHOULD NOT trigger a Sieve script invocation.

It is also possible to introduce loops through the "redirect" or "notify" actions. See Section 10 of Sieve [RFC5228], Section 8 of Sieve Notify [RFC5435], and the Security Considerations sections of the applicable notification-method documents for loop-prevention information. This extension does not change any of that advice.

Other security considerations are discussed in IMAP [RFC3501], and Sieve [RFC5228], as well as in some of the other extension documents.

7. IANA Considerations

7.1. Registration of "imapsieve" IMAP capability

IANA is asked to add "IMAPSIEVE=" to the IMAP 4 Capabilities registry, according to the IMAP 4 specification [RFC3501]. (http://www.iana.org/assignments/imap4-capabilities)

7.2. Registration of "imapsieve" Sieve extension

The following information should be added to the Sieve Extensions registry, according to the Sieve specification [RFC5228]. (http://www.iana.org/assignments/sieve-extensions/sieve-extensions.xml)

To: iana@iana.org
Subject: Registration of new Sieve extension
Capability name: imapsieve
Description: Add Sieve processing for IMAP events.
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>

7.3. Registration of Sieve Environment Items

The following subsections register items in the Sieve Environment Items registry, according to the Environment extension [RFC5183]. (http://www.iana.org/assignments/sieve-environment-items/ sieve-environment-items.xml)

7.3.1. Registration of Sieve Environment Item: cause

To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: cause
Description: The name of the action that caused the script to be invoked. Its value is one of the following:

- o APPEND (for invocations resulting from APPEND or MULTIAPPEND)

- o COPY (for invocations resulting from COPY)

- o FLAG (for invocations resulting from flag changes)

RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>

7.3.2. Registration of Sieve Environment Item: mailbox

```
To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: mailbox
Description: The name of the mailbox that the affected message is in,
in the case of existing messages, or is targeted to be stored into,
in the case of new messages.  The value of this item is fixed when
the script begins, and, in particular, MUST NOT change as a result of
any action, such as "fileinto".
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>
```

7.3.3. Registration of Sieve Environment Item: changedflags

```
To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: changedflags
Description: If the script was invoked because of flag changes to an
existing message, this contains the name(s) of the flag(s) that have
changed.  Otherwise, the value of this item MUST be the empty string.
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>
```

7.3.4. Registration of Sieve Environment Item: imapuser

```
To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: imapuser
Description: The identity (IMAP login ID) of the IMAP user that
caused the action.
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>
```

7.3.5. Registration of Sieve Environment Item: imapemail

```
To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: imapemail
Description: The primary email address of the IMAP user that caused
the action (the user identified by "imapuser").
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>
```

7.4. Registration of IMAP METADATA Mailbox Entry Name

The following information should be added to the IMAP METADATA Mailbox Entry Registry, according to the Metadata extension

[RFC5464].
(http://www.iana.org/assignments/imap-metadata/imap-metadata.xml)

To: iana@iana.org
Subject: IMAP METADATA Entry Registration
Type: Mailbox
Name: /shared/imapsieve/script
Description: This entry name is used to define mailbox metadata associated with IMAP events in Sieve for the associated mailbox. Specifically, this specifies the Sieve script that will be invoked when IMAP events occur on the specified mailbox.
Content-type: text/plain; charset=utf-8
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>

7.5. Registration of IMAP METADATA Server Entry Name

The following information should be added to the IMAP METADATA Server Entry Registry, items according to the Metadata extension [RFC5464].
(http://www.iana.org/assignments/imap-metadata/imap-metadata.xml)

To: iana@iana.org
Subject: IMAP METADATA Entry Registration
Type: Server
Name: /shared/imapsieve/script
Description: This entry name is used to define metadata associated globally with IMAP events in Sieve for the associated server. Specifically, this specifies the Sieve script that will be invoked when IMAP events occur on any mailbox in the server that does not have its own mailbox-level /shared/imapsieve/script entry.
Content-type: text/plain; charset=utf-8
RFC number: [[this RFC]]
Contact address: Sieve mailing list <sieve@ietf.org>

8. References

8.1. Normative References

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

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003.

[RFC3502] Crispin, M., "Internet Message Access Protocol (IMAP) - MULTIAPPEND Extension", RFC 3502, March 2003.

[RFC3894] Degener, J., "Sieve Extension: Copying Without Side Effects", RFC 3894, October 2004.

[RFC5183] Freed, N., "Sieve Email Filtering: Environment Extension", RFC 5183, May 2008.

[RFC5228] Guenther, P. and T. Showalter, "Sieve: An Email Filtering Language", RFC 5228, January 2008.

[RFC5232] Melnikov, A., "Sieve Email Filtering: Imap4flags Extension", RFC 5232, January 2008.

[RFC5464] Daboo, C., "The IMAP METADATA Extension", RFC 5464, February 2009.

[RFC5804] Melnikov, A. and T. Martin, "A Protocol for Remotely Managing Sieve Scripts", RFC 5804, July 2010.

[RFC6409] Gellens, R. and J. Klensin, "Message Submission for Mail", STD 72, RFC 6409, November 2011.

8.2. Informative References

[RFC4315] Crispin, M., "Internet Message Access Protocol (IMAP) - UIDPLUS extension", RFC 4315, December 2005.

[RFC5230] Showalter, T. and N. Freed, "Sieve Email Filtering: Vacation Extension", RFC 5230, January 2008.

[RFC5235] Daboo, C., "Sieve Email Filtering: Spamtest and Virustest Extensions", RFC 5235, January 2008.

[RFC5293] Degener, J. and P. Guenther, "Sieve Email Filtering: Editheader Extension", RFC 5293, August 2008.

[RFC5429] Stone, A., "Sieve Email Filtering: Reject and Extended Reject Extensions", RFC 5429, March 2009.

[RFC5435] Melnikov, A., Leiba, B., Segmuller, W., and T. Martin, "Sieve Email Filtering: Extension for Notifications", RFC 5435, January 2009.

[RFC5703] Hansen, T. and C. Daboo, "Sieve Email Filtering: MIME Part Tests, Iteration, Extraction, Replacement, and Enclosure", RFC 5703, October 2009.

Author's Address

Barry Leiba
Huawei Technologies

Phone: +1 646 827 0648
Email: barryleiba@computer.org
URI: http://internetmessagingtechnology.org/