Sieve Working Group	B. Leiba	
Internet-Draft	Huawei Technologies	
Intended status: Standards Track	September 27, 2010	
Expires: March 31, 2011		

TOC

Support for Sieve in Internet Message Access Protocol (IMAP4)
draft-ietf-sieve-imap-sieve-01

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).

Note

This document defines extensions to IMAP and Sieve. It is the work of the Sieve Working Group, but had previously been in the lemonade mailing list, as draft-ietf-lemonade-imap-sieve.

1. Discussion of this document should be taken to the Sieve mailing list at mailto:sieve@ietf.org

2. Subscription requests can be sent to mailto:sieve@ietf.org?body=subscribe (send an email message with the word "subscribe" in the body).

3. A WWW archive of back messages is available at http://www.ietf.org/mail-archive/web/sieve/index.html

4. Older messages, which were posted to the lemonade mailing list, are archived at http://www.ietf.org/mail-archive/web/lemonade/index.html

Status of this Memo

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 31, 2011.

Table of Contents

---

1. Introduction

TOC

1.1. Overview

TOC

Some applications have a need to apply [Sieve] (Guenther, P., Ed. and T. Showalter, Ed., “Sieve: An Email Filtering Language,” January 2008.) filters in situations 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] (Melnikov, A., Ed., Leiba, B., Ed., Segmuller, W., and T. Martin, “Sieve Email Filtering: Extension for Notifications,” January 2009.)).

This specification defines extensions to [IMAP] (Crispin, M., “Internet Message Access Protocol - Version 4rev1,” March 2003.) 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 IMAPSieve requires support for [Metadata] (Daboo, C., “The IMAP METADATA Extension,” February 2009.) as well, since the latter is used to associate scripts with IMAP mailboxes.
[anchor1] (General note: Sieve was designed to work at final delivery, and makes many assumptions about the context. Will those assumptions break this environment without our realizing it fully?)
[anchor2] (Note about identity: We might want to use Sieve to impose fine-grained access controls. In final delivery, there's no identity for the “filer”. Here, there is: the logged-in IMAP user. How do we get at that identity?)

1.2. Conventions used in this document

TOC

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [Keywds] (Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” March 1997.).

---

2. The IMAP "IMAPSieve" Extension

TOC

2.1. The "IMAPSieve" Capability String

TOC

An IMAP server advertises support for this extension through the capability string "IMAPSieve" (the string is not case-sensitive, and is shown here with this capitalization for readability). A server that advertises IMAPSieve is claiming to be in compliance with this specification in all aspects.
Implementations that support IMAPSieve MUST also support [Environment] (Freed, N., “Sieve Email Filtering: Environment Extension,” May 2008.), because the latter defines an important way for Sieve scripts to test the conditions under which they have been invoked. Notwithstanding this requirement, scripts that use IMAPSieve must include BOTH capability strings in their required lists.

2.2. Existing IMAP Functions Affected by IMAPSieve

TOC

The subsections below describe in detail the IMAP commands and situations on which IMAPSieve has an effect. Not all Sieve actions make sense in the case of messages affected by IMAP commands. See Section 3 (Applicable Sieve Actions and Interactions) for details.
It's important to note that since the base Sieve specification (see [Sieve] (Guenther, P., Ed. and T. Showalter, Ed., "Sieve: An Email Filtering Language," January 2008.)) 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

TOC

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 (Changes to Metadata)).

2.2.2. The IMAP MULTIAPPEND Command

TOC

If the IMAP server supports the IMAP [MultiAppend] (Crispin, M., "Internet Message Access Protocol (IMAP) - MULTIAPPEND Extension," March 2003.) extension, 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

TOC

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

TOC

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] (Melnikov, A., "Sieve Mail Filtering Language: IMAP flag Extension," January 2008.) extension 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 another script invocation. 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.

2.2.5. New or Changed IMAP Message Annotations

TOC

[anchor3] (Sieve has no way to get the annotations, so is there really value in being told about annotation changes here? Maybe push that into a sieve-annotations extension later.)
If the IMAP server supports the [Annotate] (Daboo, C. and R. Gellens, "IMAP ANNOTATE Extension," June 2008.) extension, one or more existing messages can have annotations added or changed through the ANNOTATE command. In a server that advertises IMAPSieve, messages getting new or changed annotations MUST trigger the execution of a Sieve script, subject to the settings defined through Metadata.
For annotation-change events, the Sieve script will see the message annotations as they are AFTER the changes.

2.3. New Functions Defined by IMAPSieve

TOC

2.3.1. Changes to Metadata

TOC

Support for IMAPSieve requires support for [Metadata] (Daboo, C., “The IMAP METADATA Extension,” February 2009.) 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 "/IMAPSieve/Script" for the mailbox, that entry is used. If there is not, but there is an IMAP metadata entry named "/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 an "/IMAPSieve/Script" metadata entry was selected above, the shared value of that metadata name (its "value.shared" attribute) MUST be the name of the Sieve script that will be invoked in response to the IMAP event OR the name of another metadata entry, the name prefixed with "metadata:" (such as "metadata:/IMAPSieve/ScriptContents"), that contains the actual script in its value.shared attribute. Note that only the value.shared attribute is used; any value.priv attributes are ignored.

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 MAY use [Manage] (Melnikov, A., Ed. and T. Martin, “A Protocol for Remotely Managing Sieve Scripts,” January 2009.) to acomplish this, using the PUTSCRIPT command to store the script without using the SETACTIVE command to activate it. In any case, the script name that is specified in the /IMAPSieve/Script metadata entry is in a form that depends upon how the server handles the storing of Sieve scripts.

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 extensions 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

TOC

Since some Sieve actions relate specifically to the delivery of mail, not all actions 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

TOC

For all cases that fall under IMAPSieve, 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 or annotation changes, the message is left in the mailbox.

3.2. The Keep Action

TOC

The keep action is applicable in all cases that fall under IMAPSieve. Its behaviour is as described for implicit keep, in Section 3.1 (The Implicit Keep).

3.3. The Fileinto Action

TOC

If the Sieve implementation supports the fileinto action, that action is applicable in all cases that fall under IMAPSieve. If the [Copy] (Degener, J., “Sieve Extension: Copying Without Side Effects,” October 2004.) extension 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 or annotation changes, the message is COPIED into the fileinto mailbox, without removing 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), or it MAY 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 [UIDPlus] (Crispin, M., "Internet Message Access Protocol (IMAP) - UIDPLUS Extension," December 2005.)) 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

TOC

[anchor4] (Redirect assumes message can be submitted as is – not a valid assumption in this context. What do we do if the decision is "redirect" and there's not enough information to do it? Also, some have been concerned about, say, a flag change that has the Sieve effect of forwarding the message somewhere. Perhaps we should just forbid redirect.)

The redirect action is applicable in all cases that fall under IMAPSieve. It causes the message to be sent, as specified in the base Sieve specification, to the designated address. If the [Copy] (Degener, J., "Sieve Extension: Copying Without Side Effects," October 2004.) extension 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.

For APPEND, MULTIAPPEND, and COPY, the message is stored into the target mailbox in addition to being redirected. For flag or annotation 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), or it MAY 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 [UIDPlus] (Crispin, M., "Internet Message Access Protocol (IMAP) - UIDPLUS Extension," December 2005.)) 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.

TOC

3.5. The Reject Action

The reject action is NOT applicable to any case that falls under IMAPSieve. Its use MUST result in an error condition that will terminate the Sieve script.

3.6. The Discard Action

TOC

The discard action is applicable in all cases that fall under IMAPSieve. 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), or it MAY 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 [UIDPlus] (Crispin, M., "Internet Message Access Protocol (IMAP) - UIDPLUS Extension," December 2005.)) 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.

3.7. The Notify Action

TOC

If the [Notify] (Melnikov, A., Ed., Leiba, B., Ed., Segmuller, W., and T. Martin, "Sieve Email Filtering: Extension for Notifications," January 2009.) extension is available, the notify action is applicable in all cases that fall under IMAPSieve. The result is that the requested notification is sent, and that the message is otherwise handled as it would normally have been.

3.8. The Addheader and Deleteheader Actions

TOC

[anchor5] (Should editheader be allowed to change header fields that aren't saved in place, such as for redirect or fileinto? Editheader would still have to be banned for "keep", but not otherwise.)
Even if the [EditHeader] (Degener, J. and P. Guenther, "Sieve Email Filtering: Editheader Extension," August 2008.) extension is available, since messages in IMAP mailboxes are immutable these actions are NOT applicable. Use of these MUST result in an error condition that will

terminate the Sieve script. Explanation: Using them for flag or annotation changes to existing messages would cause the message to be changed. Using them for APPEND, MULTIAPPEND, and COPY would cause unexpected differences in the stored copy as compared to what the client expected, and would make the client's message cache invalid unexpectedly.

3.9. The Setflag, Deleteflag, and Removeflag Actions

TOC

[anchor6] (Should this just require imap4flags? Some implementors have said they wouldn't bother with it without the ability to manipulate flags. And what values of flags does it see -- before or after the change? If it changes them, can it see the originals? Can it reset changes?)

If the [IMAP4Flags] (Melnikov, A., “Sieve Mail Filtering Language: IMAP flag Extension,” January 2008.) extension is available, the actions associated with it are all applicable to any case that falls under IMAPSieve. 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.

3.10. The Vacation Action

TOC

Even if the [Vacation] (Showalter, T. and N. Freed, Ed., “Sieve Email Filtering: Vacation Extension,” January 2008.) extension is available, the vacation action is NOT applicable to any case that falls under IMAPSieve. Its use MUST result in an error condition that will terminate the Sieve script.

3.11. Spamtest

TOC

[Spamtest] (Daboo, C., “Sieve Email Filtering: Spamtest and Virustest Extensions,” January 2008.) [anchor7] (We need to say something about the spamtest/virustest extension. We need to be able to scan appended messages. And we can't use headers to communicate spam status, because the message is immutable. What should we say here?)

3.12. New Sieve Environment Item: cause

TOC

Implementations MAY invoke different Sieve scripts for the different conditions described in this document (append, copy, flag changes, annotation changes). If the actions to be taken are common, and the implementation supports the [Include] (Daboo, C. and A. Stone, "SIEVE Email Filtering: Include Extension," July 2009.) extension, the common script code can be included as specified there.
It may be preferable, though, to use a single script for all these conditions. To support that, the implementation MUST set the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) item "cause", which contains the name of the action that caused the script to be invoked. Its value is one of the following:

*APPEND (for invocations resulting from APPEND or MULTIAPPEND)

*COPY (for invocations resulting from COPY)

*FLAG (for invocations resulting from flag changes)

*ANNOTATE (for invocations resulting from new or changed annotations)

3.13. New Sieve Environment Item: mailbox

TOC

The implementation MUST set the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) 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".

3.14. New Sieve Environment Item: changedflags

TOC

If the [IMAP4Flags] (Melnikov, A., "Sieve Mail Filtering Language: IMAP flag Extension," January 2008.) extension is available, AND the script was invoked because of flag changes to an existing message, the implementation MUST set the [Environment] (Freed, N., "Sieve Email

Filtering: Environment Extension," May 2008.) 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 4 (Examples) for an example of how this might be used.

3.15. New Sieve Environment Item: changedannotations

TOC

If the [Annotate] (Daboo, C. and R. Gellens, "IMAP ANNOTATE Extension," June 2008.) extension is available, AND the script was invoked because of annotation changes to an existing message, the implementation MUST set the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) item "changedannotations" to the name(s) of the annotation(s) that have changed. If the script was not invoked because of annotation changes, the value of this item MUST be the empty string.

3.16. Interaction With Sieve Tests (Comparisons)

TOC

This extension does not affect the operation of any tests or comparisons.

---

4. Examples

TOC

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"];

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] (Melnikov, A., "Sieve Mail Filtering Language: IMAP flag Extension," January 2008.) extension), a

notification is sent using the [Notify] (Melnikov, A., Ed., Leiba, B., Ed., Segmuller, W., and T. Martin, "Sieve Email Filtering: Extension for Notifications," January 2009.) extension. No notification will be sent, though, if we're called with an existing message that already had that flag set.

```
require ["notify", "imap4flags", "variables", "environment"];

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

if allof (hasflag "\\Flagged",
          not environment :contains "changedflags" "\\Flagged") {
    notify :message "Important message in ${mailbox}";
}
```

---

5. Security Considerations

TOC

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] (Melnikov, A., "Sieve Mail Filtering Language: IMAP flag Extension," January 2008.) extension. 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. Other security considerations are discussed in [IMAP] (Crispin, M., "Internet Message Access Protocol - Version 4rev1," March 2003.), and [Sieve] (Guenther, P., Ed. and T. Showalter, Ed., "Sieve: An Email Filtering Language," January 2008.), as well as in some of the other extension documents.

---

6. IANA Considerations

TOC

6.1. Registration of imapsieve extension

TOC

The following template specifies the IANA registration of the Sieve extension specified in this document:
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: Barry Leiba <barryleiba@computer.org>
This information should be added to the list of sieve extensions given on http://www.iana.org/assignments/sieve-extensions.

6.2. Registration of environment item: cause

TOC

The following template specifies the IANA registration of a sieve environment item specified in this document:
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:

*APPEND (for invocations resulting from APPEND or MULTIAPPEND)

*COPY (for invocations resulting from COPY)

*FLAG (for invocations resulting from flag changes)

*ANNOTATE (for invocations resulting from new or changed annotations)

RFC number: this RFC
Contact address:
Barry Leiba <barryleiba@computer.org>
This information should be added to the list of sieve environment item names given in the [Environment] (Freed, N., “Sieve Email Filtering: Environment Extension,” May 2008.) extension.

6.3. Registration of environment item: mailbox

TOC

The following template specifies the IANA registration of a sieve environment item specified in this document:
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:
    Barry Leiba <barryleiba@computer.org>
This information should be added to the list of sieve environment item names given in the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) extension.

6.4. Registration of environment item: changedflags

TOC

The following template specifies the IANA registration of a sieve environment item specified in this document:
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:
    Barry Leiba <barryleiba@computer.org>
This information should be added to the list of sieve environment item names given in the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) extension.

6.5. Registration of environment item: changedannotations

TOC

The following template specifies the IANA registration of a sieve environment item specified in this document:
To: iana@iana.org
Subject: Registration of new Sieve environment item
Item name: changedannotations
Description: If the script was invoked because of annotation changes to an existing message, this contains the name(s) of the annotation(s) that have changed. Otherwise, the value of this item MUST be the empty string.
RFC number: this RFC
Contact address:
    Barry Leiba <barryleiba@computer.org>
This information should be added to the list of sieve environment item names given in the [Environment] (Freed, N., "Sieve Email Filtering: Environment Extension," May 2008.) extension.

6.6. Registration of IMAP METADATA mailbox entry name

TOC

The following template specifies the IANA registration of an IMAP METADATA entry name specified in this document:
To: iana@iana.org
Subject: IMAP METADATA Registration
Please register the following IMAP METADATA item:
[x] Entry [ ] Attribute
[x] Mailbox [ ] Server
Name: /IMAPSieve/Script
Description: This entry name is used to define mailbox metadata associated with IMAPSieve events 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 person: Barry Leiba
Contact email: barryleiba@computer.org
This information should be added to the list of IMAP METADATA item names given in the [Metadata] (Daboo, C., "The IMAP METADATA Extension," February 2009.) extension.

6.7. Registration of IMAP METADATA server entry name

TOC

The following template specifies the IANA registration of an IMAP METADATA entry name specified in this document:
To: iana@iana.org
Subject: IMAP METADATA Registration
Please register the following IMAP METADATA item:
[x] Entry [ ] Attribute
[ ] Mailbox [x] Server
Name: /IMAPSieve/Script
Description: This entry name is used to define metadata associated globally with IMAPSieve events 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 /IMAPSieve/Script entry.
Content-type: text/plain; charset=utf-8
RFC number: this RFC
Contact person: Barry Leiba
Contact email: barryleiba@computer.org
This information should be added to the list of IMAP METADATA item names given in the [Metadata] (Daboo, C., "The IMAP METADATA Extension," February 2009.) extension.

7. References

TOC

7.1. Normative References

TOC

[Annotate]	Daboo, C. and R. Gellens, “IMAP ANNOTATE Extension,” RFC 5257, June 2008.
[Copy]	Degener, J., “Sieve Extension: Copying Without Side Effects,” RFC 3894, October 2004.
[Environment]	Freed, N., “Sieve Email Filtering: Environment Extension,” RFC 5183, May 2008.
[IMAP]	Crispin, M., “Internet Message Access Protocol - Version 4rev1,” RFC 3501, March 2003.
[IMAP4Flags]	Melnikov, A., “Sieve Mail Filtering Language: IMAP flag Extension,” RFC 5232, January 2008.
[Keywds]	Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” RFC 2119, March 1997.
[Metadata]	Daboo, C., “The IMAP METADATA Extension,” RFC 5464, February 2009.
[MultiAppend]	Crispin, M., “Internet Message Access Protocol (IMAP) - MULTIAPPEND Extension,” RFC 3502, March 2003.
[Sieve]	Guenther, P., Ed. and T. Showalter, Ed., “Sieve: An Email Filtering Language,” RFC 5228, January 2008.

7.2. Non-Normative References

TOC

[EditHeader]	Degener, J. and P. Guenther, “Sieve Email Filtering: Editheader Extension,” RFC 5293, August 2008.
[Include]	Daboo, C. and A. Stone, “SIEVE Email Filtering: Include Extension,” work in progress, http://tools.ietf.org/html/draft-ietf-sieve-include, July 2009.
[Manage]	Melnikov, A., Ed. and T. Martin, “A Protocol for Remotely Managing Sieve Scripts,” RFC editor queue, http://tools.ietf.org/html/draft-ietf-sieve-managesieve, January 2009.
[Notify]	Melnikov, A., Ed., Leiba, B., Ed., Segmuller, W., and T. Martin, “Sieve Email Filtering: Extension for Notifications,” RFC 5435, January 2009.
[Spamtest]	Daboo, C., “Sieve Email Filtering: Spamtest and Virustest Extensions,” RFC 5235, January 2008.
[UIDPlus]	Crispin, M., “Internet Message Access Protocol (IMAP) - UIDPLUS Extension,” RFC 4315, December 2005.

[Vacation]	Showalter, T. and N. Freed, Ed., “Sieve Email Filtering: Vacation Extension,” RFC 5230, January 2008.

Author's Address

TOC

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