Workgroup: EXTRA
Internet-Draft:
draft-ietf-extra-sieve-mailboxid-05
U 5228 (if approved)
p
d
a
t
e
s
:
Published: 19 November 2020
Intended Status: Standards Track
Expires: 23 May 2021
A B. Gondwana, Ed.
uFastmail
t
h
o
r
s
:

Sieve Email Filtering: delivery by mailboxid

Abstract

The OBJECTID capability of the IMAP protocol (RFC8474) allows clients to identify mailboxes by a unique identifier which survives rename.

This document extends the Sieve mail filtering language (RFC5228) to allow using that same unique identifier as a target for fileinto rules, and for testing the existance of mailboxes.

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 https://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 23 May 2021.

Table of Contents

1. Introduction

[RFC5228] Sieve rules are sometimes created using graphical interfaces which allow users to select the mailbox to be used as a target for a rule.

If that mailbox is renamed, the client may also update its internal representation of the rule and update the sieve script to match, however this is a multi-step process and subject to partial failures. Also, if the folder is renamed by a different mechanism (e.g. another IMAP client) the rules will get out of sync.

By telling "fileinto" to reference the immutable mailboxid specified by [RFC8474], using the extension specified herein, sieve rules can continue to target the same mailbox even if it gets renamed.

2. Conventions Used In This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Sieve capability string

Scripts which use the following extensions MUST explicitly require the capability "mailboxid".

Example:

```
require "mailboxid";
```

4. Argument ":mailboxid" to Command "fileinto"

Normally, the "fileinto" command delivers the message in the mailbox specified using its positional mailbox argument. However, if the optional ":mailboxid" argument is also specified, the "fileinto" command first checks whether a mailbox exists in the user's personal namespace [RFC2342] with the specified [RFC8474] MAILBOXID.

If a matching mailbox is found, that mailbox is used for delivery.

If there is no such mailbox, the "fileinto" action proceeds as it would without the ":mailboxid" argument.

The tagged argument :mailboxid to fileinto consumes one additional token, a string with the objectid of the mailbox to file into.

Example:

```
require "fileinto";
require "mailboxid";

if header :contains ["from"] "coyote" {
    fileinto :mailboxid "F6352ae03-b7f5-463c-896f-d8b48ee3"
             "INBOX.harassment";
}
```

4.1. Interaction with "mailbox" extension

For servers which also support the [!@RFC5490] mailbox extension, the ":create" modifier to fileinto does not create the mailbox with the specified mailboxid, however it may be specified and interacts as normal with all other extensions.

Example:

```
require "fileinto";
require "mailboxid";
require "mailbox";

fileinto :mailboxid "Fnosuch"
         :create
         "INBOX.no-such-folder";
            # creates INBOX.no-such-folder, but it doesn't
            # get the "Fnosuch" mailboxid.
```

4.2. Interaction with "specialuse" extension

For servers which also support [!@RFC8579] delivery to special-use mailboxes, it is an error to specify both ":mailboxid" and ":special-use" in the same fileinto command.

Advanced filtering based on both special-use and mailboxid can be built with explicit "specialuse_exists" and "mailboxidexists" tests.

Note to developers of sieve generation tools: it is advisable to use special-use rather than mailboxid when creating rules that are based on a special-use purpose (e.g. delivery directly to the Junk folder based on a header that was added by a scanning agent earlier in the mailflow).

5. Interaction with "fcc" extension

This document extends the definition of the ":fcc" argument defined in [RFC8580] so that it can optionally be used with the ":mailboxid" argument.

```
FCC =/ [":mailboxid" <mailboxid: string>]
```

If the optional ":mailboxid" argument is specified with ":fcc", it instructs the Sieve interpreter to check whether a mailbox exists with the specific mailboxid. If such a mailbox exists, the generated message is filed into that mailbox. Otherwise, the generated message is filed into the ":fcc" target mailbox.

Example:

```
require ["enotify", "fcc", "mailboxid"];
notify :fcc "INBOX.Sent"
       :mailboxid "F6352ae03-b7f5-463c-896f-d8b48ee3"
       :message "You got mail!"
       "mailto:ken@example.com";
```

6. Test "mailboxidexists"

The "mailboxidexists" test is true if all mailboxes listed in the "mailboxids" argument exist in the mailstore, and each allows the user in whose context the Sieve script runs to "deliver" messages into it. When the mailstore is an IMAP server, "delivery" of messages is possible if:

a) the READ-WRITE response code is present for the mailbox (see Section 7.1 of [RFC3501]), if IMAP Access Control List (ACL) [RFC4314] is not supported by the server, or

b) the user has 'p' or 'i' rights for the mailbox (see Section 5.2 of [RFC4314]).

Note that a successful "mailboxidexists" test for a mailbox doesn't necessarily mean that a "fileinto :mailboxid" action on this mailbox would succeed. For example, the "fileinto" action might put user over quota. The "mailboxidexists" test only verifies existence of the mailbox and whether the user in whose context the Sieve script runs has permissions to execute "fileinto" on it.

Example:

```
require "fileinto";
require "mailboxid";

if header :contains ["from"] "coyote" {
    if mailboxidexists "F6352ae03-b7f5-463c-896f-d8b48ee3" {
        fileinto :mailboxid "F6352ae03-b7f5-463c-896f-d8b48ee3"
                            "INBOX.harassment";
    } else {
        fileinto "INBOX.harassment";
    }
}
```

Note to implementers: this test behaves identically to the mailboxexists test defined in [RFC5490] but operates on mailboxids rather than mailbox names.

7. Interaction with variables extension

There is no special interaction defined, however as an objectid is a string in this document, objectid values can contain variable expansions if [RFC5229] is enabled.

8. Formal Syntax

```
test =/ "mailboxidexists" string-list

tag =/ ":mailboxid" string

FCC =/ [":mailboxid" <mailboxid: string>]
       ; if [@RFC8580] is supported
```

9. Security considerations

Because mailboxid is always generated by the server, implementations MUST NOT allow sieve to make an endrun around this protection by creating mailboxes with the specified ID by using ":create" and ":mailboxid" in a fileinto rule for a non-existant mailbox.

Implementers are referred to the security considerations sections of [RFC5228] and [RFC8474].

10. IANA considerations

IANA are requested to add a capability to the sieve-extensions registry:

```
To: iana@iana.org
Subject: Registration of new Sieve extension

Capability name: mailboxid
Description: adds a test for checking mailbox existence by objectid,
             and new optional arguments to fileinto and :fcc which
             allow selecting the destination mailbox by objectid.
RFC number: this RFC
Contact address: The EXTRA discussion list <extra@ietf.org>
```

11. Acknowledgements

This document borrows heavily from [RFC5490] for the matching mailboxexists test, and from [RFC8579] for an example of modifying the fileinto command.

Thanks to Ned Freed and Ken Murchison and Alexey Melnikov for feedback on the EXTRA mailing list.

12. Changes

(EDITOR: remove this section before publication)

12.1. draft-ietf-sieve-mailboxid-05

*disallow :mailboxid and :special-use in the same fileinto action.

12.2. draft-ietf-sieve-mailboxid-04

*made RFC5490 and RFC8579 normative

*clarified wording based on AD feedback from Barry

12.3. draft-ietf-sieve-mailboxid-03

*Fixed ABNF syntax error

12.4. draft-ietf-sieve-mailboxid-02

*removed bogus : from "mailboxidexists" test title

*moved FCC to its own top-level section since it is not used with the fileinto command.

12.5. draft-ietf-sieve-mailboxid-01

*fixed idnits - RFC5228 not mentioned in the abstract

*fixed other I-D references I had missed, oops

12.6. draft-ietf-sieve-mailboxid-00

*Adopted into working group per adoption call on list

*Updated references to old drafts which have since been published.

*Fixed some typoes and simplified some language.

*Removed stray leading colon on mailboxexists (thanks Alexey)

*Added :fcc to the IANA registration description (thanks Alexey)

*Mentioned that variables can be expanded (thanks Alexey)

12.7. draft-gondwana-sieve-mailboxid-02

*Update document date by a couple of years! Ooops, it got forgotten after a WGLC which got not dissent.

*Create xml2rfc v3 output.

12.8. draft-gondwana-sieve-mailboxid-01

*Switch to :mailboxid tagged parameter value with fallback mailbox name.

*Document interaction with "mailbox".

*Document interaction with "special-use".

*Document interaction with "fcc".

*Document security considerations around :mailboxid and :create.

12.9. draft-gondwana-sieve-mailboxid-00

*Initial version.

13. Normative References

[RFC2342] Gahrns, M. and C. Newman, "IMAP4 Namespace", RFC 2342, DOI 10.17487/RFC2342, May 1998, <https://www.rfc-editor.org/info/rfc2342>.

[RFC8580] Murchison, K. and B. Gondwana, "Sieve Extension: File Carbon Copy (FCC)", RFC 8580, DOI 10.17487/RFC8580, May 2019, <https://www.rfc-editor.org/info/rfc8580>.

[RFC8474] Gondwana, B., Ed., "IMAP Extension for Object Identifiers", RFC 8474, DOI 10.17487/RFC8474, September 2018, <https://www.rfc-editor.org/info/rfc8474>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC5228] Guenther, P., Ed. and T. Showalter, Ed., "Sieve: An Email Filtering Language", RFC 5228, DOI 10.17487/RFC5228, January 2008, <https://www.rfc-editor.org/info/rfc5228>.

14. Informative References

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, DOI 10.17487/RFC3501, March 2003, <https://www.rfc-editor.org/info/rfc3501>.

[RFC8579] Bosch, S., "Sieve Email Filtering: Delivering to Special-Use Mailboxes", RFC 8579, DOI 10.17487/RFC8579, May 2019, <https://www.rfc-editor.org/info/rfc8579>.

[RFC4314] Melnikov, A., "IMAP4 Access Control List (ACL) Extension", RFC 4314, DOI 10.17487/RFC4314, December 2005, <https://www.rfc-editor.org/info/rfc4314>.

[RFC5490] Melnikov, A., "The Sieve Mail-Filtering Language -- Extensions for Checking Mailbox Status and Accessing Mailbox Metadata", RFC 5490, DOI 10.17487/RFC5490, March 2009, <https://www.rfc-editor.org/info/rfc5490>.

[RFC5229] Homme, K., "Sieve Email Filtering: Variables Extension", RFC 5229, DOI 10.17487/RFC5229, January 2008, <https://www.rfc-editor.org/info/rfc5229>.

Author's Address

Bron Gondwana (editor)
Fastmail
Level 2, 114 William St
Melbourne VIC 3000
Australia

Email: brong@fastmailteam.com
URI: https://www.fastmail.com