

Workgroup: EXTRA  
Internet-Draft: draft-murchison-email-snooze-00  
Published: 10 July 2023  
Intended Status: Standards Track  
Expires: 11 January 2024  
Authors: K. Murchison R. Signes N. Jenkins  
Fastmail Fastmail Fastmail  
**Snoozing Email with IMAP, JMAP, and Sieve**

## Abstract

This document describes the "snooze" extensions to IMAP, JMAP for Mail, and the Sieve Email Filtering Language. The "snooze" extensions give these protocols the ability to postpone the appearance of an email message in a target mailbox until a later point in time.

## 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 11 January 2024.

## Copyright Notice

Copyright (c) 2023 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 (<https://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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

## Table of Contents

- [1. Introduction](#)
  - [1.1. Conventions Used in This Document](#)
- [2. Mechanics of Snoozing and Awakening a Message](#)
- [3. IMAP SNOOZE Extension](#)
  - [3.1. The \Snoozed Mailbox Attribute](#)
  - [3.2. SNOOZE Command](#)
  - [3.3. UID SNOOZE Command](#)
  - [3.4. Semantics of SNOOZE and UID SNOOZE](#)
  - [3.5. Interaction with Other Extensions](#)
    - [3.5.1. ANNOTATE](#)
  - [3.6. Formal Syntax](#)
- [4. JMAP Mail Snooze Extension](#)
  - [4.1. The "snoozed" Mailbox Role](#)
  - [4.2. Extension to the Email Object](#)
- [5. Sieve Snooze Extension](#)
  - [5.1. Snooze Action](#)
    - [5.1.1. Mailbox Argument](#)
    - [5.1.2. Times, TZID, and Weekdays Arguments](#)
    - [5.1.3. Interaction with Extensions to the Fileinto Action](#)
- [6. Implementation Status](#)
- [7. Security Considerations](#)
- [8. Privacy Considerations](#)
- [9. IANA Considerations](#)
  - [9.1. Registration of JMAP Capability](#)
  - [9.2. Registration of IMAP Capability](#)
  - [9.3. Registration of Sieve Extension](#)
  - [9.4. Registration of Sieve Action](#)
  - [9.5. Registration of IMAP Mailbox Name Attribute](#)
  - [9.6. Registration of IMAP Annotation Entry](#)
- [10. Acknowledgments](#)
- [11. References](#)
  - [11.1. Normative References](#)
  - [11.2. Informative References](#)
- [Authors' Addresses](#)

### 1. Introduction

Users are not always ready, willing, or able to read and respond to email messages at the time of their arrival. Sometimes it is desirable to have messages appear in a mailbox at a more convenient time for the user to act upon them.

This document defines extensions to the [Internet Message Access Protocol \(IMAP\) \[RFC9051\]](#), [The JSON Meta Application Protocol \(JMAP\) for Mail \[RFC8621\]](#), and the [Sieve Email Filtering Language \[RFC5228\]](#) that enable postponing the appearance of a message in a target

mailbox until a later point in time, also known as "snoozing" a message.

The IMAP and JMAP extensions allow for snoozing messages that are already present in the mailstore. The Sieve extension allows for snoozing a message at the time of final delivery.

A "snoozed" message is one that the user wants put into a less obtrusive location for a fixed time. When that time is reached, the message is "awakened" back into higher visibility.

### **1.1. 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.

## **2. Mechanics of Snoozing and Awakening a Message**

The extensions defined in this specification use the following procedure for snoozing and awakening messages:

1. The message is delivered to, or moved into, a "snoozed" mailbox in the mailstore.
2. The message is assigned three meta-data items:
  - \*An awaken time - the time at which to move the message out of the "snoozed" mailbox.
  - \*An optional target mailbox into which the message will be moved at the awaken time.
  - \*An optional set of flag/keyword updates to be made when the message is awakened.
3. Messages MUST be accessible while in the "snoozed" mailbox.
4. At the awaken time, the message is moved from the "snoozed" mailbox into the target mailbox, and any updates are made to the message flags/keywords. If the target mailbox is not specified, or is not a valid or available target for the user, then the message is moved into the user's main mailbox.

### 3. IMAP SNOOZE Extension

The SNOOZE extension is present in any IMAP implementation that returns "SNOOZE" as one of the supported capabilities to the CAPABILITY command.

#### 3.1. The \Snoozed Mailbox Attribute

Users may create a mailbox with an attribute of [\Snoozed](#) ([Section 9.5](#)), having the sole purpose of holding snoozed emails. This mailbox is exposed over JMAP as having the "snoozed" role.

Implementations are REQUIRED to enforce the following restrictions on the \Snoozed mailbox:

\*Messages MUST NOT be added to the \Snoozed mailbox with the APPEND, COPY, or MOVE commands. The server MUST reject commands that attempt to use the \Snoozed mailbox as a target with a tagged NO response having the CANNOT response code.

\*When executing COPY or MOVE commands on the \Snoozed mailbox, the server MUST NOT copy "snoozed" meta-data to the new message. The new message is effectively awakened.

#### 3.2. SNOOZE Command

**Arguments:** sequence-set

date/time string

OPTIONAL add flag list

OPTIONAL remove flag list

OPTIONAL mailbox name

**Responses:** no specific responses for this command

**Result:** OK - snooze completed

NO - snooze error: can't snooze those messages

BAD - command unknown or arguments invalid

#### 3.3. UID SNOOZE Command

This extends the first form of the UID command (see [[RFC9051](#)], [Section 6.4.9](#)) to add the SNOOZE command defined above as a valid argument.

### 3.4. Semantics of SNOOZE and UID SNOOZE

The SNOOZE command takes two mandatory arguments: a message set (sequence numbers for SNOOZE, UIDs for UID SNOOZE) and a date/time string (awaken time); and three optional arguments: a named target mailbox, a list of flags to be added to the awakened messages, and a list of flags to be removed from the awakened messages. Each message included in the set is moved (as per [\[RFC9051\]](#), [Section 6.4.9](#)) from the selected mailbox to the mailbox having the \Snoozed attribute. Additionally, the awaken time, any target mailbox name, and any flag lists are stored as "snoozed" meta-data on the messages. This process appears to the client as a single action and has the same effect for each message as this sequence (using syntax from [\[RFC5257\]](#), [Section 4.5](#)):

1. [UID] STORE ANNOTATION (/snoozed (value.shared ...))
2. [UID] MOVE

At the awaken time, each message is moved from the \Snoozed mailbox into the target mailbox, and any updates are made to the message flags. If no target mailbox was specified or it is inaccessible, then the message is moved into the user's INBOX. and has the same effect for each message as this sequence:

1. [UID] STORE +FLAGS.SILENT
2. [UID] STORE -FLAGS.SILENT
3. [UID] MOVE

### 3.5. Interaction with Other Extensions

Because of the similarity of SNOOZE to MOVE, extensions that affect MOVE affect SNOOZE in the same way. Response codes listed in [\[RFC9051\]](#), [Section 7.1](#), as well as those defined by extensions, are sent as indicated for MOVE.

#### 3.5.1. ANNOTATE

Servers supporting [ANNOTATE](#) [\[RFC5257\]](#) MAY expose the "snoozed" meta-data on messages as a read-only shared [/snoozed](#) ([Section 9.5](#)) entry. The value of the [/snoozed](#) entry MUST be a JSON string in the form of a [snoozeDetails object](#) ([Section 4.2](#)).

### 3.6. Formal Syntax

The following syntax specification uses the Augmented Backus-Naur Form (ABNF) notation as specified in [\[RFC5234\]](#). [\[RFC9051\]](#) defines the

non-terminals "capability", "command-select", "sequence-set", "mailbox", "date-time", and "flag-list".

Except as noted otherwise, all alphabetic characters are case insensitive. The use of upper or lower case characters to define token strings is for editorial clarity only. Implementations MUST accept these strings in a case-insensitive fashion.

capability        =/ "SNOOZE"

command-select =/ snooze

snooze           = "SNOOZE" SP sequence-set SP date-time  
                  [ SP "+FLAGS" SP flag-list ]  
                  [ SP "-FLAGS" SP flag-list ]  
                  [ SP mailbox ]

uid              = "UID" SP  
                  (copy / move / fetch / search / store /  
                  uid-expunge / snooze)

#### 4. JMAP Mail Snooze Extension

The capabilities object is returned as part of the JMAP Session object; see [[RFC8620](#)], [Section 2](#). Servers supporting this specification MUST add a property named "urn:ietf:params:jmap:mail:snooze" to the capabilities object. The value of this property is an empty object in both the JMAP session "capabilities" property and an account's "accountCapabilities" property.

##### 4.1. The "snoozed" Mailbox Role

Users may create a mailbox with a role of "snoozed", having the sole purpose of holding snoozed emails. This mailbox is exposed over IMAP as having the [Snoozed](#) ([Section 9.5](#)) attribute.

##### 4.2. Extension to the Email Object

The Email [[RFC8621](#)], [Section 4.1](#) object gains a new property:

\***snoozed**: SnoozeDetails|null (immutable)

If in the mailbox with role snoozed, this Email will be moved from there according to the details in this object. Otherwise, this is a record of when it was snoozed in the past.

A **SnoozeDetails** object has the following properties:

\***until**: UTCDate

The time at which to move the message out of the "snoozed" mailbox.

**\*moveToMailboxId:** String (optional)

If set and a valid mailbox id for the user, this is where the Email will be moved at the awaken time. Otherwise, the Email will be moved to the user's inbox.

**\*setKeywords:** String[Boolean] (optional)

If set, when the message is awakened it will also have keywords modified. The keys of the object are the keyword names. Names that map to true are keywords to be added. Names that map to false are keywords to remove.

The **snoozed** property has the following rules and restrictions:

\*It is treated as an immutable property, i.e. it cannot be updated on an existing copy in a mailbox, it must cause a new copy (with a new UID).

\*If an Email is in more than one mailbox (i.e. the same GUID is in multiple mailboxes), **snoozed** MUST only be set on one of the copies.

\*When setting **snoozed** to a non-null value in an Email/set method:

- If the email is already in the Snoozed mailbox and not being removed from this mailbox, replace the copy in the Snoozed mailbox and set the SnoozeDetails on the new email.

- Otherwise, if adding the Email to a single new mailbox, set the SnoozeDetails on this copy.

- Otherwise, if adding the Email to multiple mailboxes, set the SnoozeDetails on the copy in:

  - oThe Snoozed mailbox if this is being added.

  - oOtherwise, the mailbox with the id in the moveToMailboxId property of the SnoozeDetails object (or inbox if this property is omitted), if this is being added.

If neither apply, reject the create/update with an `invalidProperties` error.

- If not adding the Email to a new mailbox, replace the copy that currently has SnoozeDetails. If none, reject the update with an `invalidProperties` error.

-Clear any existing SnoozeDetails from Emails in any other mailbox (this will involve replacing the copy with a new UID of course, as it's immutable).

\*When removing the mailbox with which **snoozed** is associated from an Email, the snooze is implicitly cleared (i.e. it is not implicitly copied to another copy of the message; it must be explicitly set if the client wants to preserve it).

## 5. Sieve Snooze Extension

Sieve implementations that support this extension have an identifier of "snooze" for use with the capability mechanism.

Conventions for notations are as in [Section 1.1](#) of [\[RFC5228\]](#), including use of the "Usage:" label for the definition of action and tagged arguments syntax.

### 5.1. Snooze Action

Usage: snooze \*AWAKEN-OPTIONS <times: string-list>

The AWAKEN-OPTIONS argument is defined here in [ABNF \[RFC5234\]](#) syntax so that it can be modified by other extensions.

```
AWAKEN-OPTIONS = MAILBOX / WEEKDAYS / TZID
                  ; each option MUST NOT appear more than once
                  ; however, per Section 2.6.2 of RFC 5228,
                  ; the tagged arguments in AWAKEN-OPTIONS
                  ; may appear in any order
```

```
MAILBOX = ":mailbox" string
WEEKDAYS = ":weekdays" string-list
TZID     = ":tzid" string
```

The "snooze" action cancels the implicit keep and postpones delivery of the message into the specified mailbox at a later point in time.

The snooze action is semantically equivalent to a delayed fileinto action (see [Section 4.1](#) of [\[RFC5228\]](#)). The arguments of the snooze action specify when, where, and how the awakened message will be filed.

Implementations MUST snooze and awaken messages per [Section 2](#). Specifically, a Sieve interpreter whose mailstore is accessible via IMAP and/or JMAP MUST utilize a "snoozed" mailbox per [Sections Section 3](#) and [Section 4](#) respectively.



### 5.1.1. Mailbox Argument

The optional `:mailbox` argument is used to specify the target mailbox that the message will be filed into when it is awakened. It is equivalent to the `mailbox` argument of the `fileinto` action (see [Section 4.1](#) of [[RFC5228](#)]).

If `:mailbox` is omitted, or if the specified mailbox doesn't exist at the time of awakening, the message will be filed into the user's main mailbox. For instance, in an implementation where an IMAP server is running scripts on behalf of the user at time of delivery, the user's "INBOX" would be the implicit target for awakening messages.

### 5.1.2. Times, TZID, and Weekdays Arguments

The required `times` argument, along with the optional `:tzid` and `:weekdays` arguments, are used to specify when a snoozed message will be awakened. Each time is specified in "hh:mm:ss" format and is interpreted as the local time in the time zone specified by the `:tzid` argument.

The value of the `:tzid` argument MUST be a time zone identifier from the [IANA Time Zone Database](#) [[tzdb](#)]. If `:tzid` is omitted, the time zone of the Sieve interpreter is used.

The `:weekdays` argument specifies the set of days on which the specified set of times apply. Each day of the week is expressed as an integer between "0" and "6". "0" is Sunday, "1" is Monday, etc. This syntax matches that of the "weekday" date-part argument to the date test extension (see [Section 4.2](#) of [[RFC5260](#)]). If `:weekdays` is omitted, the set of times applies to every day of the week.

The combination of the weekdays and times form a chronological list of awaken times. When a message is snoozed, it is assigned the next future awaken time in the list. If a message is snoozed on a day with no awaken times, or after the last awaken time on a given day, the first awaken time on the next available day is used.

If the local time in the specified time zone occurs more than once (daylight saving to standard time transition), the first occurrence of the specified time value is used. If the local time in the specified time zone does not occur (standard to daylight saving time transition), the specified time value is interpreted using the UTC offset prior to the transition.

#### 5.1.2.1. Awaken Times Examples

The following examples show, given the specified snooze action and a set of message arrival times, the corresponding times at which the message would be awakened and filed.

The following example shows awoken times rolling into the next day or week. Note that 2020-07-30 falls on a Thursday.

```
require "snooze";
snooze :weekdays ["1", "3", "5", "2", "4"]
      :tzid "Australia/Melbourne" ["12:00:00",
                                   "08:00:00", "16:00:00"];
```

Arrival (UTC)	Arrival (Melbourne)	Awaken (Melbourne)
2020-07-30T00:00:00Z	--07-30T10:00:00+10	--07-30T12:00:00+10
2020-07-30T04:00:00Z	--07-30T14:00:00+10	--07-30T16:00:00+10
2020-07-30T08:00:00Z	--07-30T18:00:00+10	--07-31T08:00:00+10
2020-07-31T12:00:00Z	--07-31T22:00:00+10	--08-03T08:00:00+10
2020-08-01T16:00:00Z	--08-02T02:00:00+10	--08-03T08:00:00+10

Table 1

The following example shows awoken times falling before, during, and after a daylight saving to standard time transition. Note that the transition occurs at 2020-11-01T02:00:00-04.

```
require "snooze";
snooze :tzid "America/New_York" "01:30:00";
```

Arrival (UTC)	Arrival (New York)	Awaken (New York)
2020-11-01T05:00:00Z	--11-01T01:00:00-04	--11-01T01:30:00-04
2020-11-01T06:00:00Z	--11-01T01:00:00-05	--11-02T01:30:00-05
2020-11-01T07:00:00Z	--11-01T02:00:00-05	--11-02T01:30:00-05

Table 2

The following example shows awoken times falling before, during, and after a standard to daylight saving time transition. Note that the transition occurs at 2021-03-14T02:00:00-05.

```
require "snooze";
snooze :tzid "America/New_York" "02:30:00";
```

Arrival (UTC)	Arrival (New York)	Awaken (New York)
2021-03-13T06:30:00Z	--03-13T01:30:00-05	--03-13T02:30:00-05
2021-03-14T06:30:00Z	--03-14T01:30:00-05	--03-14T03:30:00-04
2021-03-14T07:30:00Z	--03-14T03:30:00-04	--03-15T02:30:00-04

Table 3

### 5.1.3. Interaction with Extensions to the Fileinto Action

Some tagged arguments defined in extensions to the fileinto action can be used together with the snooze action. The sections below describe these interactions. Tagged arguments in future extensions to the fileinto action need to describe their interaction with the snooze extension, if any.

When any `fileinto` extension arguments are used with the `snooze` extension, the corresponding extension MUST be enabled, and the arguments are defined to have the same syntax, semantics, and treatment as they do with the `fileinto` action.

#### 5.1.3.1. `Imap4flags` Extension

When the `"imap4flags"` [RFC5232] extension is enabled in a script, two additional tagged arguments are added to `"snooze"` that allow manipulating the set of flags on a snoozed message.

AWAKEN-OPTIONS /= ADDFLAGS / REMOVEFLAGS

ADDFLAGS = ":addflags" string-list

REMOVEFLAGS = ":removeflags" string-list

The optional `:addflags` and `:removeflags` arguments are used to specify which `IMAP` [RFC9051] flags should be added to and/or removed from the set of `IMAP` flags present on the snoozed message at the time of awakening. Note the set of `IMAP` flags present at the time of awakening may be the empty set.

If the `"setflag"` and/or `"addflag"` actions have been used to store `IMAP` flags in the `imap4flags` internal variable, the Sieve interpreter MUST use the current value of the internal variable as the set of flags to associate with the message when storing it into the `"snoozed"` mailbox.

This document doesn't dictate how the Sieve interpreter will set the `IMAP` flags. In particular, the Sieve interpreter may work as an `IMAP` client or may have direct access to the mailstore.

The general requirements for flag handling specified in [Section 2](#) of [RFC5232] MUST be followed.

##### 5.1.3.1.1. Example

The following example leverages the `Date` [RFC5260], `Relational` [RFC5231], and `Imap4flags` [RFC5232] extensions to snooze messages received after business hours until the following work day. Note that the message is marked as `important` when it is snoozed, and will be marked as `unread` when it is awakened.

```
require ["snooze", "imap4flags", "date", "relational"];

if anyof(header :is "from" "boss@example.com",
         currentdate :is "weekday" "0",
         currentdate :is "weekday" "6",
         currentdate :value "ge" "hour" "17") {
  setflag "\\Important";
  snooze :removeflags "\\Seen"
        :weekdays ["1". "2", "3", "4", "5"]
        :tzid "American/New_York", "09:00";
}
```

### 5.1.3.2. Mailbox Extension

This document extends the definition of the [":create"](#) [RFC5490] tagged argument so that it can be used with the snooze action.

```
AWAKEN-OPTIONS /= CREATE
```

```
CREATE = ":create"
        ; MUST NOT be appear unless MAILBOX also appears
```

If the optional `":create"` argument is specified with snooze, it instructs the Sieve interpreter to create the target mailbox, if needed, before attempting to file the awakened message into the target mailbox.

### 5.1.3.3. Special-Use Extension

This document extends the definition of the [":specialuse"](#) [RFC8579] tagged argument so that it can be used with the snooze action.

```
AWAKEN-OPTIONS /= SPECIAL-USE
```

```
SPECIAL-USE = ":specialuse" string
```

If the optional `":specialuse"` argument is specified with snooze, it instructs the Sieve interpreter to check whether a mailbox exists with the specific special-use flag assigned to it. If such a mailbox exists, the awakened message is filed into the special-use mailbox. Otherwise, the awakened message is filed into the target mailbox.

If both the optional `":specialuse"` and `":create"` arguments are specified with snooze, the Sieve interpreter is instructed to create the target mailbox per [Section 4.1](#) of [RFC8579], if needed.

### 5.1.3.4. MailboxID Extension

This document extends the definition of the [":mailboxid"](#) [RFC9042] tagged argument so that it can be used with the snooze action.

AWAKEN-OPTIONS /= MAILBOXID

MAILBOXID = ":mailboxid" string

If the optional ":mailboxid" argument is specified with snooze, it instructs the Sieve interpreter to check whether a mailbox exists in the user's [personal namespace \[RFC2342\]](#) with the specified [MAILBOXID \[RFC8474\]](#). If such a mailbox exists, the awakened message is filed into that mailbox. Otherwise, the awakened message is filed into the target mailbox.

It is an error to specify both ":mailboxid" and ":specialuse" in the same snooze action.

## 6. Implementation Status

< RFC Editor: before publication please remove this section and the reference to [\[RFC7942\]](#) >

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in [\[RFC7942\]](#). The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to [\[RFC7942\]](#), "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

### 6.1. Cyrus Server

The open source [Cyrus Server](#) project is a highly scalable enterprise mail system which supports IMAP, JMAP, and Sieve email filtering at the point of final delivery. This production level implementation supports all of the requirements described in this document. This implementation is freely distributable under a BSD style license from [Computing Services at Carnegie Mellon University](#).

## 7. Security Considerations

Security considerations are discussed in [RFC5228], [RFC5232], [RFC8579], and [RFC9042].

It is believed that this extension doesn't introduce any additional security concerns.

## 8. Privacy Considerations

It is believed that this extension doesn't introduce any privacy considerations beyond those in [RFC5228].

## 9. IANA Considerations

### 9.1. Registration of JMAP Capability

This document defines the following new JMAP capability to be added to the registry defined in [Section 9.4](#) of [RFC8620] and located here: <https://www.iana.org/assignments/jmap/jmap.xhtml#jmap-capabilities>

IANA are requested to add a capability to the JMAP Capabilities registry:

Capability Name: urn:ietf:params:jmap:mail:snooze

Specification document: RFC XXXX

Intended use: common

Change Controller: IETF

Security and privacy considerations: RFC XXXX, [Section 7](#)

### 9.2. Registration of IMAP Capability

IANA are requested to add "SNOOZE" to the IMAP Capabilities registry, <https://www.iana.org/assignments/imap4-capabilities>

### 9.3. Registration of Sieve Extension

This document defines the following new Sieve extension to be added to the registry defined in [Section 6.2](#) of [RFC5228] and located here: <https://www.iana.org/assignments/sieve-extensions/sieve-extensions.xhtml#sieve-extensions>

IANA are requested to add a capability to the Sieve Extensions registry:

To: [iana@iana.org](mailto:iana@iana.org)

Subject: Registration of new Sieve extension

Capability name: snooze

Description: Adds the "snooze" action command to postpone delivery of a message into a target mailbox until a later point in time.

RFC number: RFC XXXX

Contact address: The Sieve discussion list <sieve@ietf.org>

#### 9.4. Registration of Sieve Action

This document defines the following new Sieve action to be added to the registry defined in [Section 3.1](#) of [[I-D.ietf-extra-sieve-action-registry](#)].

IANA are requested to add an action to the Sieve Action registry:

Name: snooze

Description: Postpone delivery of a message into a target mailbox until a later point in time.

References: RFC XXXX, [[RFC5232](#)], [[RFC5490](#)], [[RFC8579](#)], [[RFC9042](#)]

Capabilities: "snooze", "imap4flags", "mailbox", "special-use", "mailboxid".

Interactions: Is not compatible with the reject or ereject actions.

Cancels Implicit Keep?: Y

Use with IMAP Events?: Y

Comments: Requires a special "snoozed" mailbox in the mailstore.

#### 9.5. Registration of IMAP Mailbox Name Attribute

This document defines the following new IMAP mailbox name attribute to be added to the registry defined in [Section 6.2](#) of [[RFC8457](#)] and located here: <https://www.iana.org/assignments/imap-mailbox-name-attributes/imap-mailbox-name-attributes.xhtml#imap-mailbox-name-attributes>

IANA are requested to add an attribute to the IMAP Mailbox Name Attribute registry:

To: iana@iana.org

Subject: Registration of new IMAP Mailbox Name Attribute

Attribute name: Snoozed

Description: Messages that have been snoozed.

Reference: RFC XXXX

## 9.6. Registration of IMAP Annotation Entry

This document defines the following new IMAP annotation entry to be added to the registry defined in [Section 6](#) of [RFC5257] and located here: <https://www.iana.org/assignments/imap-annotate-extension/imap-annotate-extension.xhtml#imap-annotate-extension-entries>

To: iana@iana.org

Subject: IMAP Annotate Registration

Please register the following Annotate item:

[X]Entry            [] Attribute

Name: /snoozed

Description: RFC XXXX, [Section 3.5.1](#).

Content-Type: application/json

Contact person: Ken Murchison

email: [murch@fastmailteam.com](mailto:murch@fastmailteam.com)

## 10. Acknowledgments

The authors would like to thank the following individuals for contributing their ideas and support for writing this specification: Ned Freed, Barry Leiba, Alexey Melnikov, and Pete Resnick.

## 11. References

### 11.1. Normative References

[I-D.ietf-extra-sieve-action-registry] Melnikov, A. and K. Murchison, "IANA Registry for Sieve Actions", Work in Progress, Internet-Draft, draft-ietf-extra-sieve-action-registry-06, 27 March 2023, <<https://datatracker.ietf.org/doc/html/draft-ietf-extra-sieve-action-registry-06>>.

[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>>.
- [RFC2342] Gahrns, M. and C. Newman, "IMAP4 Namespace", RFC 2342, DOI 10.17487/RFC2342, May 1998, <<https://www.rfc-editor.org/info/rfc2342>>.
- [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>>.
- [RFC5232] Melnikov, A., "Sieve Email Filtering: Imap4flags Extension", RFC 5232, DOI 10.17487/RFC5232, January 2008, <<https://www.rfc-editor.org/info/rfc5232>>.
- [RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.
- [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>>.
- [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>>.
- [RFC8457] Leiba, B., Ed., "IMAP "\$Important" Keyword and "\Important" Special-Use Attribute", RFC 8457, DOI 10.17487/RFC8457, September 2018, <<https://www.rfc-editor.org/info/rfc8457>>.
- [RFC8474] Gondwana, B., Ed., "IMAP Extension for Object Identifiers", RFC 8474, DOI 10.17487/RFC8474, September 2018, <<https://www.rfc-editor.org/info/rfc8474>>.
- [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>>.
- [RFC8620] Jenkins, N. and C. Newman, "The JSON Meta Application Protocol (JMAP)", RFC 8620, DOI 10.17487/RFC8620, July 2019, <<https://www.rfc-editor.org/info/rfc8620>>.
- [RFC8621] Jenkins, N. and C. Newman, "The JSON Meta Application Protocol (JMAP) for Mail", RFC 8621, DOI 10.17487/RFC8621, August 2019, <<https://www.rfc-editor.org/info/rfc8621>>.

**[RFC9042]**

Gondwana, B., Ed., "Sieve Email Filtering: Delivery by MAILBOXID", RFC 9042, DOI 10.17487/RFC9042, June 2021, <<https://www.rfc-editor.org/info/rfc9042>>.

**[RFC9051]**

Melnikov, A., Ed. and B. Leiba, Ed., "Internet Message Access Protocol (IMAP) - Version 4rev2", RFC 9051, DOI 10.17487/RFC9051, August 2021, <<https://www.rfc-editor.org/info/rfc9051>>.

**[tzdb]**

Internet Assigned Numbers Authority, "Time Zone Database", <<https://www.iana.org/time-zones>>.

## 11.2. Informative References

**[RFC5231]**

Segmuller, W. and B. Leiba, "Sieve Email Filtering: Relational Extension", RFC 5231, DOI 10.17487/RFC5231, January 2008, <<https://www.rfc-editor.org/info/rfc5231>>.

**[RFC5257]**

Daboo, C. and R. Gellens, "Internet Message Access Protocol - ANNOTATE Extension", RFC 5257, DOI 10.17487/RFC5257, June 2008, <<https://www.rfc-editor.org/info/rfc5257>>.

**[RFC5260]**

Freed, N., "Sieve Email Filtering: Date and Index Extensions", RFC 5260, DOI 10.17487/RFC5260, July 2008, <<https://www.rfc-editor.org/info/rfc5260>>.

**[RFC7942]**

Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", BCP 205, RFC 7942, DOI 10.17487/RFC7942, July 2016, <<https://www.rfc-editor.org/info/rfc7942>>.

## Authors' Addresses

Kenneth Murchison  
Fastmail US LLC  
1429 Walnut Street - Suite 1201  
Philadelphia, PA 19102  
United States of America

Email: [murch@fastmailteam.com](mailto:murch@fastmailteam.com)

Ricardo Signes  
Fastmail US LLC  
1429 Walnut Street - Suite 1201  
Philadelphia, PA 19102  
United States of America

Email: [rjbs@fastmailteam.com](mailto:rjbs@fastmailteam.com)

Neil Jenkins  
Fastmail Pty Ltd  
Level 2, 114 William Street  
Melbourne VIC 3000  
Australia

Email: [neilj@fastmailteam.com](mailto:neilj@fastmailteam.com)