

EXTRA
Internet-Draft
Intended status: Standards Track
Expires: November 5, 2018

S. Bosch
Dovecot Oy
May 4, 2018

Internet Message Access Protocol (IMAP) - SAVEDATE Extension
draft-ietf-extra-imap-savedate-01

Abstract

This document adds a new capability called "SAVEDATE" to the Internet Message Access Protocol (IMAP). It defines a new IMAP message attribute called 'save date' that, unlike the existing 'internal date' attribute, always indicates the moment at which the message was saved in its current mailbox. The SAVEDATE capability extends the FETCH command with the means to retrieve the save date attribute and it extends the SEARCH command to allow using that attribute in searching criteria.

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 November 5, 2018.

Copyright Notice

Copyright (c) 2018 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	2
2.	Conventions Used in This Document	3
3.	Save Date Message Attribute	3
4.	IMAP Protocol Changes	4
4.1.	CAPABILITY Identification	4
4.2.	FETCH Command and Response Extensions	4
4.3.	SEARCH Command Extension	4
5.	Formal Syntax	5
6.	Security Considerations	5
7.	IANA Considerations	5
8.	Acknowledgements	6
9.	Normative References	6
	Author's Address	6

[1.](#) Introduction

This document extends Internet Message Access Protocol (IMAP) [[IMAP4rev1](#)] with a new capability called "SAVEDATE". This capability adds a new IMAP message attribute called 'save date'. The save date is the date and time at which a message was saved in the mailbox it currently resides. The save date is similar to the existing internal date attribute in that it is set at the time of delivery. However, the internal date attribute can be set to an arbitrary value for messages delivered to the mailbox using the APPEND command and it is usually copied from the source message for messages delivered using the COPY command. In contrast, the save date attribute is always set to the current date and time at the moment the message is saved in the mailbox, irrespective of how the message is delivered and from where it originates.

The save date message attribute is useful for implementing automated removal of messages from a mailbox after a configured amount of time. For that application, it is necessary to know when the message was saved in the mailbox, which cannot be reliably determined using the the internal date attribute.

For example, a common client usage pattern is to move deleted messages to a Trash mailbox; these messages are considered "deleted" at the time they are moved to the Trash mailbox. In an effort to limit the size of the Trash mailbox, a client may subsequently desire to permanently remove (expunge) all messages in that Trash mailbox deleted before a certain time (e.g. a configurable expiration

Bosch

Expires November 5, 2018

[Page 2]

interval). In that case, the internal date attribute cannot be used, since it likely refers to the time at which the message was originally received. The proper time comparison attribute should instead be the time at which the message was saved to the Trash mailbox. Similar usage patterns can be observed for Archiving solutions.

The SAVEDATE capability extends the FETCH command and response to allow retrieving the date and time at which a message was saved. Also, the SAVEDATE capability extends the SEARCH command to allow searching messages with criteria based on when it was saved in the mailbox.

2. Conventions Used in This Document

In examples, "C:" indicates lines sent by a client that is connected to a server. "S:" indicates lines sent by the server to the client.

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 [[KEYWORDS](#)].

3. Save Date Message Attribute

The save date message attribute is the date and time at which the message was saved in the mailbox it is now located in. Unlike the internal date attribute defined by [[IMAP4rev1](#)], this date and time value cannot be set arbitrarily when a message is delivered by the IMAP APPEND command. Also, when a message is delivered to the mailbox by the IMAP COPY command [[IMAP4rev1](#)] or the IMAP MOVE command [[IMAP-MOVE](#)], the save date attribute is not copied from the source message. Instead, the current date and time at which the message is delivered to a mailbox MUST be used to set the save date attribute. Once calculated, the save date attribute MUST NOT change as long as the message is contained within the same mailbox.

This means that when the message is copied to another mailbox, the save date of the message in source mailbox remains unaffected; only the new copy of the message gets a new save date. Consequently, when the message is moved to another mailbox, either using the MOVE command or a command sequence involving the COPY command [[IMAP-MOVE](#)], the message always gets a new save date in the destination mailbox.

For some specific mailboxes, the underlying storage may not support the save date message attribute. The handling of this situation is described in detail in the next section for each involved IMAP command.

4. IMAP Protocol Changes

4.1. CAPABILITY Identification

IMAP servers that support this extension MUST include "SAVEDATE" in the response list to the CAPABILITY command.

4.2. FETCH Command and Response Extensions

This extension defines one new data item for the FETCH command:

SAVEDATE

The save date of the message.

This extension defines one new data item for the FETCH response:

SAVEDATE

A string representing the save date of the message. However, if the underlying mailbox storage does not support the save date message attribute, the value returned for the SAVEDATE item is always NIL, rather than a string.

Example:

```
C: A101 FETCH 998 (SAVEDATE)
S: * 998 FETCH (SAVEDATE "01-Jan-2015 18:50:53 +0100")
S: A101 OK Fetch completed.
```

4.3. SEARCH Command Extension

This extension defines four new search keys for the SEARCH command:

SAVEDBEFORE <date>

Messages whose save date (disregarding time and timezone) is earlier than the specified date.

SAVEDON <date>

Messages whose save date (disregarding time and timezone) is within the specified date.

SAVEDSINCE <date>

Messages whose save date (disregarding time and timezone) is within or later than the specified date.

SAVEDATESUPPORTED

Matches all messages in the mailbox when the underlying storage of that mailbox supports the save date attribute. Conversely, it

Bosch

Expires November 5, 2018

[Page 4]

matches no messages in the mailbox when the save date attribute is not supported.

When the underlying storage of a mailbox does not support the save date attribute, the SAVEDBEFORE, SAVEDON, and SAVEDSINCE search keys MUST use the internal date attribute instead. In the context of the IMAP Multimapbox SEARCH Extension [[MULTISEARCH](#)], this fallback behavior MUST apply to each mailbox individually, meaning that only the mailboxes that lack support for the save date attribute use the internal date attribute instead.

Example:

```
C: A102 SEARCH SAVEDON 28-Dec-2014
S: * SEARCH 993 994
S: A102 OK Search completed.
C: A103 SEARCH SAVEDSINCE 28-Dec-2014
S: * SEARCH 993 994 995 996 997 998 999 1000 1001
S: A103 OK Search completed.
```

5. Formal Syntax

The following syntax specification augments the grammar specified in [[IMAP4rev1](#)]. It uses the Augmented Backus-Naur Form (ABNF) notation as specified in [[ABNF](#)]. Elements not defined here are taken from [[IMAP4rev1](#)].

```
capability      =/ "SAVEDATE"

fetch-att       =/ "SAVEDATE"
msg-att-static  =/ "SAVEDATE" SP (date-time / nil)

search-key      =/ "SAVEDBEFORE" SP date /
                  "SAVEDON" SP date /
                  "SAVEDSINCE" SP date /
                  "SAVEDATESUPPORTED"
```

6. Security Considerations

There are no known additional security issues with this extension beyond those described in the base protocol described in [[IMAP4rev1](#)].

7. IANA Considerations

The IANA is requested to add "SAVEDATE" to the "IMAP Capabilities" registry located at <<http://www.iana.org/assignments/imap-capabilities>>.

8. Acknowledgements

Thanks to Bron Gondwana, Alexey Melnikov, Timo Sirainen, and Michael Slusarz for reviews and suggestions.

9. Normative References

- [ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.
- [IMAP-MOVE] Gulbrandsen, A. and N. Freed, "Internet Message Access Protocol (IMAP) - MOVE Extension", [RFC 6851](#), January 2013.
- [IMAP4rev1] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), March 2003.
- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [MULTISEARCH] Leiba, B. and A. Melnikov, "IMAP4 Multimailbox SEARCH Extension", [RFC 7377](#), DOI 10.17487/RFC7377, October 2014, <<https://www.rfc-editor.org/info/rfc7377>>.

Author's Address

Stephan Bosch
Dovecot Oy
Lars Sonckin Kaari 10
Espoo 02600
Finland

Email: stephan.bosch@dovecot.fi

