

Workgroup: Network Working Group
Internet-Draft:
draft-ietf-extra-imap-uidonly-00
Published: 30 January 2023
Intended Status: Standards Track
Expires: 3 August 2023
Authors: A. Melnikov A. P. Achuthan V. Nagulakonda
 Isode Yahoo! Yahoo!
 A. Singh L. Alves
 Yahoo!

IMAP Extension for only using and returning UIDs

Abstract

The UIDONLY extension of the Internet Message Access Protocol (RFC 3501/RFC 9051) allows clients to request that all information about mailbox changes is returned only using UIDs. Message numbers are not going to be returned and can't be used once this extension is enabled. This helps both clients and servers to reduce resource usage required for maintenance of message number to UID map.

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 3 August 2023.

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.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

Table of Contents

- [1. Introduction and Overview](#)
- [2. Document Conventions](#)
- [3. The UIDONLY extension](#)
 - [3.1. Enabling UIDONLY extension](#)
 - [3.2. Changes to FETCH/STORE/SEARCH](#)
 - [3.3. Changes to UID FETCH/UID STORE](#)
 - [3.4. Changes to EXPUNGE/UID EXPUNGE](#)
 - [3.5. Changes to how other mailbox changes are announced](#)
- [4. Formal syntax](#)
- [5. Security Considerations](#)
- [6. IANA Considerations](#)
 - [6.1. Changes/additions to the IMAP4 capabilities registry](#)
- [7. Acknowledgments](#)
- [8. Normative References](#)
- [Authors' Addresses](#)

1. Introduction and Overview

This document defines an extension to the Internet Message Access Protocol [[RFC3501](#)] for eliminating use of message numbers. This extension is compatible with both IMAP4rev1 [[RFC3501](#)] and IMAP4rev2 [[RFC9051](#)].

The UIDONLY extension of the Internet Message Access Protocol (RFC 3501/RFC 9051) allows clients to request that servers only use and return UIDs. This helps both clients and servers to reduce resource usage required for maintenance of message number to UID map.

IMPLEMENTATION NOTE: this document is not yet at the state where it is implementable. Please contact document authors if you want to experiment with implementing UIDONLY.

2. Document Conventions

In protocol examples, this document uses a prefix of "C: " to denote lines sent by the client to the server, and "S: " for lines sent by the server to the client. Lines prefixed with "// " are comments explaining the previous protocol line. These prefixes and comments are not part of the protocol. Lines without any of these prefixes are continuations of the previous line, and no line break is present in the protocol unless specifically mentioned.

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.

Other capitalised words are IMAP keywords [[RFC3501](#)][[RFC9051](#)] or keywords from this document.

3. The UIDONLY extension

An IMAP server advertises support for the UIDONLY extension by including the "UIDONLY" capability in the CAPABILITY response/response code.

The UIDONLY extension affects how information about new, expunged or changed messages is returned in unsolicited responses. In particular, it affects responses to UID FETCH/UID STORE/EXPUNGE/UID EXPUNGE, as well as how unsolicited mailbox changes are announced.

The following subsections describe changes introduced by this extension in more details.

3.1. Enabling UIDONLY extension

As the UIDONLY extension affects how information about new, expunged or changed messages is returned in unsolicited responses, it has to be enabled by the client first using the ENABLE command.

```
S: * OK [CAPABILITY IMAP4rev1 ENABLE CONDSTORE QRESYNC UIDONLY AUTH=SC
C: 01 ENABLE UIDONLY
S: * ENABLED UIDONLY
S: 01 OK ENABLE completed
```

3.2. Changes to FETCH/STORE/SEARCH

Is this Ok or is this too restrictive? Should COPY/MOVE be restricted in the same way?

When UIDONLY is enabled, FETCH, STORE and SEARCH commands are prohibited and MUST result in a tagged BAD response. Clients should instead use UID FETCH, UID STORE and UID SEARCH respectively.

3.3. Changes to UID FETCH/UID STORE

When UIDONLY is enabled, all FETCH responses that would be returned by UID FETCH/UID STORE are replaced by UIDFETCH responses.

Note that UIDFETCH response contains the same FETCH response data items, except the UID, which is returned differently at the beginning of a UIDFETCH response. Requesting UID FETCH data item is not an error and this is just ignored.

```
C: 10 UID FETCH 25900:26600 (FLAGS)
[...]
S: * 25996 UIDFETCH (FLAGS (\Seen))
S: * 25997 UIDFETCH (FLAGS (\Flagged \Answered))
S: * 26600 UIDFETCH (FLAGS ())
S: 10 OK FETCH completed
```

```
C: 11 UID FETCH 25900:26600 (UID FLAGS)
S: * 25900 UIDFETCH (FLAGS (\Seen))
S: * 25902 UIDFETCH (FLAGS (\Flagged))
S: * 26310 UIDFETCH (FLAGS (\Answered))
S: * 26311 UIDFETCH (FLAGS ())
S: * 26498 UIDFETCH (FLAGS (\Answered))
[...]
S: 11 OK FETCH completed
```

3.4. Changes to EXPUNGE/UID EXPUNGE

When UIDONLY is enabled, all EXPUNGED responses that would be returned by EXPUNGE/UID EXPUNGE are replaced by VANISHED responses, as defined in [\[RFC7162\]](#). Note that a server that implements UIDONLY extension is not required to also implement CONDSTORE and/or QRESYNC extensions.

```
C: 12 EXPUNGE
S: * VANISHED 405,407,410,425
S: 12 OK expunged
```

3.5. Changes to how other mailbox changes are announced

When UIDONLY is enabled, all changes to flags (and other dynamic message attributes) are returned using UIDFETCH responses, instead of FETCH responses.

Similarly, all expunged messages are announced using VANISHED responses instead of EXPUNGE responses.

UID MOVE/UID COPY commands SHOULD return COPYUID response code, as specified in [[RFC4315](#)].

Use of UIDNOTSTICKY response code (see [[RFC4315](#)]) is not compatible with the UIDONLY extension. I.e. a server that advertises UIDONLY extension MUST NOT return UIDNOTSTICKY response code.

Also need to think about interaction with CONDSTORE/QRESYNC in more details. CONDSTORE is compatible with UIDONLY, but QRESYNC might not be, due to existence of message numbers in SELECT QRESYNC.

4. Formal syntax

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

Non-terminals referenced but not defined below are as defined by [IMAP4](#) [[RFC3501](#)].

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.

```
SP                = <Defined in RFC 5234>

capability        =/ "UIDONLY"
                  ;; <capability> from [RFC3501]

message-data     =/ uidfetch-resp

uidfetch-resp    = uniqueid SP "UIDFETCH" SP msg-att
                  ;; UID msg-att is never used. It is replaced by th
                  ;; at the beginning of the UIDFETCH response

message-data     =/ expunged-resp

expunged-resp    = <defines VANISHED response, see RFC 7162>
```

5. Security Considerations

TBD.

6. IANA Considerations

6.1. Changes/additions to the IMAP4 capabilities registry

IMAP4 capabilities are registered by publishing a standards track or IESG approved Informational or Experimental RFC. The registry is currently located at:

<https://www.iana.org/assignments/imap4-capabilities>

IANA is requested to add definition of the UIDONLY extension to point to this document.

7. Acknowledgments

TBD.

8. Normative References

- [ABNF] Crocker, D., Ed. and P. Overell, Ed., "Augmented BNF for Syntax Specifications: ABNF", RFC 5234, January 2008, <<https://www.rfc-editor.org/info/rfc5234>>.
- [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>>.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, DOI 10.17487/RFC3501, March 2003, <<https://www.rfc-editor.org/info/rfc3501>>.
- [RFC4315] Crispin, M., "Internet Message Access Protocol (IMAP) - UIDPLUS extension", RFC 4315, DOI 10.17487/RFC4315, December 2005, <<https://www.rfc-editor.org/info/rfc4315>>.
- [RFC7162] Melnikov, A. and D. Cridland, "IMAP Extensions: Quick Flag Changes Resynchronization (CONDSTORE) and Quick Mailbox Resynchronization (QRESYNC)", RFC 7162, DOI 10.17487/RFC7162, May 2014, <<https://www.rfc-editor.org/info/rfc7162>>.
- [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>>.
- [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>>.

Authors' Addresses

Alexey Melnikov
Isode Limited

Email: alexey.melnikov@isode.com
URI: <https://www.isode.com>

Arun Prakash Achuthan
Yahoo Inc.

Email: arunprakash@myyahoo.com

Vikram Nagulakonda
Yahoo Inc.

Email: nvikram_imap@yahoo.com

Ashutosh Singh
Yahoo Inc.

Email: ashutoshvsingh@yahoo.com

Luis Alves

Email: luis.alves@lafaspot.com