

Network Working Group

Internet-Draft

Updates: [3501](#), [4466](#), [2088](#) (if approved)

Intended status: Standards Track

Expires: May 4, 2017

A. Melnikov

Isode Ltd

J. SrimushnamBoovaraghamoorthy

Samsung Electronics America

October 31, 2016

64bit body part and message sizes in IMAP4

[draft-melnikov-imap-64bit-01.txt](#)

Abstract

This document defines an IMAPv4rev1 extension that extends the existing IMAPv4rev1 32 Bit message and body part sizes to 63 bit.

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 May 4, 2017.

Copyright Notice

Copyright (c) 2016 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.

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	2
2.	Requirements Notation	2
3.	64bit Extension	3
4.	IMAP Protocol Changes	3
5.	Examples	3
6.	Formal Syntax	3
7.	Security Considerations	4
8.	IANA Considerations	5
9.	Acknowledgments	5
10.	Normative References	5
	Authors' Addresses	5

[1.](#) Introduction

IMAP [[RFC3501](#)] only allows body parts or message sizes which are 32 bit. This document introduces an IMAP extension that allows for message and body part sizes to be 63 bit.

The client wishing to use this extension MUST issue ENABLE 64BIT. Refer [[RFC5161](#)] for the usage of ENABLE command.

[2.](#) Requirements Notation

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 [[RFC2119](#)].

In examples, "C:" and "S:" indicate lines sent by the client and server respectively. If a single "C:" or "S:" label applies to multiple lines, then the line breaks between those lines are for editorial clarity only and are not part of the actual protocol exchange.

3. 64bit Extension

An IMAP server that supports the 64bit extension advertises this by including the name 64BIT in its CAPABILITY list in the authenticated state. The server may also advertise this extension before the user has logged in. If this capability is omitted, no information is conveyed about the server's status of supporting this extension.

IMAP server should respond with BAD response for the 64bit message size messages sent by the IMAP client unless it issues "ENABLE 64BIT" in the current connection.

4. IMAP Protocol Changes

TBD.

5. Examples

```
C: t1 CAPABILITY
S: * CAPABILITY IMAP4rev1 ID 64BIT
S: t1 OK foo
C: t2 ENABLE 64BIT
S: * ENABLED 64BIT
S: t2 OK foo
```

6. 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 [\[RFC3501\]](#).

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.


```
body-extension  =/ number64
                  ; Alexey: I am not sure if this change is absolutely needed!

body-fld-lines   = number64

body-fld-octets  = number64

fetch-att        =/ "BODY" section ["<" number64 "." nz-number64 ">"] /
                  "BODY.PEEK" section ["<" number64 "." nz-number64 ">"]

literal          = "{" number64 ["+"] "}" CRLF *CHAR8
                  ; number64 represents the number of CHAR8s.
                  ; NOTE: "+" can only present when LITERAL+/LITERAL-
                  ; is also advertised

literal8         = "~{" number64 ["+"] "}" CRLF *OCTET
                  ;; Updating RFC 4466 version.
                  ;; A string that might contain NULs.
                  ;; <number> represents the number of OCTETs
                  ;; in the response string.
                  ;; The "+" is only allowed when both LITERAL+/LITERAL-
                  ;; and BINARY extensions are supported by the server
                  ;; [RFC2088] - needs to be updated on LITERAL-
                  ;; publication

msg-att-static   =/ "RFC822.SIZE" SP number64

search-key       =/ "LARGER" SP number64 / "SMALLER" SP number64

number64         = 1*DIGIT
                  ; Unsigned 63-bit integer
                  ; (0 <= n <= 9,223,372,036,854,775,807)

nz-number64      = digit-nz *DIGIT
                  ; Unsigned 63-bit integer
                  ; (0 < n <= 9,223,372,036,854,775,807)

CHAR8            = <defined in RFC 3501>
```

7. Security Considerations

TBD.

This document doesn't raise any other security concerns not already raised by [[RFC3501](#)].

8. IANA Considerations

IANA is asked to add "64BIT" to the IMAP Capabilities registry, using this document as its reference.

9. Acknowledgments

TBD.

10. Normative References

- [ABNF] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.
- [RFC2088] Myers, J., "IMAP4 non-synchronizing literals", [RFC 2088](#), DOI 10.17487/RFC2088, January 1997, <<http://www.rfc-editor.org/info/rfc2088>>.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), DOI 10.17487/RFC3501, March 2003, <<http://www.rfc-editor.org/info/rfc3501>>.
- [RFC3516] Nerenberg, L., "IMAP4 Binary Content Extension", [RFC 3516](#), DOI 10.17487/RFC3516, April 2003, <<http://www.rfc-editor.org/info/rfc3516>>.
- [RFC4466] Melnikov, A. and C. Daboo, "Collected Extensions to IMAP4 ABNF", [RFC 4466](#), DOI 10.17487/RFC4466, April 2006, <<http://www.rfc-editor.org/info/rfc4466>>.
- [RFC5161] Gulbrandsen, A., Ed. and A. Melnikov, Ed., "The IMAP ENABLE Extension", [RFC 5161](#), DOI 10.17487/RFC5161, March 2008, <<http://www.rfc-editor.org/info/rfc5161>>.

Authors' Addresses

Alexey Melnikov
Isode Ltd
14 Castle Mews
Hampton, Middlesex TW12 2NP
UK

Email: Alexey.Melnikov@isode.com

Jayantheesh SrimushnamBoovaraghamoorthy
Samsung Electronics America
685 US Highway 202/206
Bridgewater, New Jersey 08807
USA

Email: jayantheesh.sb@gmail.com