Internet Engineering Task Force (IETF)
Request for Comments: 8438
Category: Standards Track
ISSN: 2070-1721

S. Bosch
Dovecot Oy
August 2018

IMAP Extension for STATUS=SIZE

Abstract

This document adds a new capability called "STATUS=SIZE" to the Internet Message Access Protocol (IMAP). It allows retrieving the total storage size of a mailbox with a single STATUS command rather than retrieving and summing the sizes of all individual messages in that mailbox.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8438.

Copyright Notice

Table of Contents

1. Introduction

This document extends the Internet Message Access Protocol (IMAP) [IMAP4rev1] with a new capability called "STATUS=SIZE". To determine the total storage size of a mailbox, an IMAP client currently needs to retrieve all message sizes individually using the FETCH command with the RFC822.SIZE data item. The STATUS=SIZE capability provides a more efficient means of achieving this. It extends the STATUS command with a new "SIZE" data item, which indicates the total size of all messages in the target mailbox. This way, this information can be queried with just one STATUS command. When the LIST-STATUS IMAP capability [LIST-STATUS] is also available, the SIZE data item can be queried for many mailboxes at once using just one LIST command.

This capability is particularly useful for IMAP clients that do not cache the state of the message store, such as most webmail clients. Without the "STATUS=SIZE" capability, such clients need to fetch all message sizes from the server when the size of an individual mailbox needs to be determined. For example, a user may request detailed quota usage information for each mailbox to find out which specific mailboxes consume most of the available storage resources. Using this information, the user can get an overview of which mailboxes need to be cleaned up to reduce quota usage. The QUOTA capability [QUOTA] is no help in that scenario, since the provided QUOTAROOT command can only yield the STORAGE resource usage of a whole quota root, not each individual mailbox within that root.

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", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [KEYWORDS] [KEYWORDS2] when, and only when, they appear in all capitals, as shown here.

3. STATUS Command and Response Extensions

This extension defines one new status data item for the STATUS command and response:

SIZE

The total size of the mailbox in octets. This is not strictly required to be an exact value, but it MUST be equal to or greater than the sum of the values of the RFC822.SIZE FETCH message data item [IMAP4rev1] of all messages in the mailbox. When the QUOTA capability [QUOTA] is also supported, this value SHOULD be equal to the storage usage value used to enforce the STORAGE resource limit for this mailbox. This way, the client can directly infer the quota usage.

Since the total storage size of a mailbox can easily exceed 4 GB, clients MUST be capable of receiving 63-bit SIZE data item values. The message size is chosen to be at most 63 bits wide rather than 64 bits to make implementations on various platforms (such as Java) easier.

Example:

```
C: A01 STATUS frop (MESSAGES SIZE UIDNEXT)
S: * STATUS frop (MESSAGES 8 SIZE 44421 UIDNEXT 242344)
S: A01 OK STATUS completed
```

The same information can be obtained by using the following commands, albeit less efficiently:

```
C: A02 SELECT "frop"
S: * FLAGS (\Answered \Flagged \Deleted \Seen \Draft)
S: * 8 EXISTS
S: * 1 RECENT
S: * OK [UNSEEN 7] First unseen.
S: * OK [UIDVALIDITY 1364851417] UIDs valid
S: * OK [UIDNEXT 242344] Predicted next UID
S: * OK [HIGHESTMODSEQ 3914] Highest
S: A02 OK [READ-WRITE] Select completed.
C: A03 FETCH 1:* (RFC822.SIZE)
S: * 1 FETCH (RFC822.SIZE 3224)
S: * 2 FETCH (RFC822.SIZE 1222)
S: * 3 FETCH (RFC822.SIZE 444)
S: * 4 FETCH (RFC822.SIZE 4516)
S: * 5 FETCH (RFC822.SIZE 544)
S: * 6 FETCH (RFC822.SIZE 922)
S: * 7 FETCH (RFC822.SIZE 31126)
S: * 8 FETCH (RFC822.SIZE 2423)
S: A03 OK Fetch completed.
```

When the LIST-STATUS IMAP capability [LIST-STATUS] is also available, the STATUS command can be combined with the LIST command to further improve efficiency. This way, the sizes of many mailboxes can be queried with just one LIST command.

Example:

```
C: A04 LIST "" % RETURN (STATUS (MESSAGES SIZE))
S: * LIST () "." "INBOX"
S: * STATUS "INBOX" (MESSAGES 17 SIZE 16234)
S: * LIST () "." "frop"
S: * STATUS "frop" (MESSAGES 8 SIZE 44421)
S: A04 OK List completed.
```

4. Formal Syntax

The following syntax specification augments the grammar specified in [IMAP4rev1] and [IMAP4-ABNF]. It uses the Augmented Backus-Naur Form (ABNF) notation as specified in [ABNF]. Elements not defined here are taken from [IMAP4rev1] and [IMAP4-ABNF].

```
capability     =/ "STATUS=SIZE"

status-att     =/ "SIZE"

status-att-val =/ "SIZE" SP number64

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

5. Security Considerations

There are no known additional security issues with this extension beyond those described for the base protocol [IMAP4rev1] and the LIST-STATUS extension [LIST-STATUS].

6. IANA Considerations

IANA has added "STATUS=SIZE" to the "IMAP Capabilities" registry located at <http://www.iana.org/assignments/imap-capabilities>.

7. Normative References

[ABNF] 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>.

[IMAP4-ABNF] Melnikov, A. and C. Daboo, "Collected Extensions to IMAP4 ABNF", RFC 4466, DOI 10.17487/RFC4466, April 2006, <https://www.rfc-editor.org/info/rfc4466>.

[IMAP4rev1] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, DOI 10.17487/RFC3501, March 2003, <https://www.rfc-editor.org/info/rfc3501>.

[KEYWORDS] 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>.

[KEYWORDS2] 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>.

[LIST-STATUS] Melnikov, A. and T. Sirainen, "IMAP4 Extension for Returning STATUS Information in Extended LIST", RFC 5819, DOI 10.17487/RFC5819, March 2010, <https://www.rfc-editor.org/info/rfc5819>.

[QUOTA] Myers, J., "IMAP4 QUOTA extension", RFC 2087, DOI 10.17487/RFC2087, January 1997, <https://www.rfc-editor.org/info/rfc2087>.

Acknowledgements

Thanks to Bron Gondwana, Alexey Melnikov, Stan Kalisch, and Michael Slusarz for reviews and suggestions.

Author's Address

Stephan Bosch
Dovecot Oy
Lars Sonckin Kaari 12
Espoo 02600
Finland

Email: stephan.bosch@dovecot.fi