

Message ORGanization Working Group
Internet-Draft
Intended status: Standards Track
Expires: March 26, 2011

A. Melnikov
Isode Limited
September 22, 2010

IMAP4 POSTADDRESS extension
draft-melnikov-imap-postaddress-07

Abstract

The POSTADDRESS extension of the Internet Message Access Protocol ([RFC 3501](#)) permits a client to discover an email address that can be used to send messages to a user's IMAP mailbox.

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 March 26, 2011.

Copyright Notice

Copyright (c) 2010 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

Internet-Draft

IMAP4 POSTADDRESS extension

September 2010

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.	Conventions used in this document	3
2.	Introduction and Overview	3
2.1.	Comparison of POSTADDRESS with URLAUTH/BURL	3
3.	LIST command with the POSTADDRESS parameter	4
4.	Extended LIST response with POSTADDRESS information	6
5.	Formal Syntax	7
6.	Security Considerations	7
7.	IANA Considerations	8
8.	Acknowledgements	9
9.	References	9
9.1.	Normative References	9
9.2.	Informative References	9
	Author's Address	10

1. 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.

In all examples "/" character is used as hierarchy separator.

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 [RFC 2119](#) [[KEYWORDS](#)].

2. Introduction and Overview

IMAP POSTADDRESS extension can be used to discover an email address for a given IMAP mailbox. Many email clients support saving a copy of an outgoing message in "sent messages" or "outbox" mailbox. Typically, those email clients send the message first using SMTP. After that they upload a copy of the message using IMAP APPEND. Effectively, the message is sent twice: once using SMTP and once using IMAP. If the IMAP server supports the POSTADDRESS extension, the mail client can avoid uploading a copy of the message using IMAP APPEND. This can be achieved by specifying an additional SMTP recipient, returned by LIST RETURN (POSTADDRESS) command, during submission.

Note: Similar functionality can be provided when the message to be sent out is first uploaded with IMAP APPEND command to the IMAP server and then submitted using IMAP [[RFC4467](#)] and SMTP [[RFC4468](#)] extensions. See [section 2.1](#) for the detailed comparison between POSTADDRESS and URLAUTH/BURL.

A server that supports POSTADDRESS parameter to the LIST command MUST return "POSTADDRESS" in its capability response. Any server supporting the POSTADDRESS extension defined in this document MUST also support the LIST-EXTENDED extension defined in [[LISTEXT](#)].

[2.1.](#) Comparison of POSTADDRESS with URLAUTH/BURL

Note that the POSTADDRESS extension is easier to implement on the server than the combination of IMAP [[RFC4467](#)] and SMTP [[RFC4468](#)] extensions.

There are certain situations when the POSTADDRESS extension provides functionality not otherwise available with URLAUTH/BURL:

1. The POSTADDRESS extension can be used when message assembly is done during submission using multiple BDAT [[RFC3030](#)] and/or BURL

Melnikov

Expires March 26, 2011

[Page 3]

Internet-Draft

IMAP4 POSTADDRESS extension

September 2010

[RFC4468] commands. URLAUTH/BURL requires that a fully assembled message is first uploaded/created in an IMAP mailbox.

2. The POSTADDRESS extension can be used to save a copy of the message in multiple email accounts on one or different server, or in the IMAP mailbox located on a different IMAP server. For example, a user might have access to different IMAP accounts in her client, but would like to save all messages she sent in a mailbox on one of the servers.
3. A POSTADDRESS email address can be used as the envelope MAIL FROM address (to capture bounces), or as the [RFC2822](#) From address, for subscribing to mailing lists, etc.

There are some performance related advantages of using POSTADDRESS:

If all the message data is generated locally, use of APPEND/GENURLAUTH [[RFC4467](#)] takes 2 round-trips:

C: IMAP APPEND

S: ...returns the UID of the appended message...

C: GENURLAUTH for the appended message (using URL constructed from the UID)

S: URL to be used in BURL

Some clients may require an additional round-trip to determine if the submission server supports BURL. This may be elided if the server advertises "BURL imap" in the first EHLO response, or if the client either assumes this to be the case, or if it has this capability

cached.

When using POSTADDRESS, the client needs to discover the posting email address for a mailbox once and can cache it after that. The two round-trips mentioned above will be saved for each submitted message. This is a plus for links with high latency.

Note, that any returned POSTADDRESS email address may be subject to user-controlled delivery filtering, such as [\[RFC5228\]](#), which may cause a message sent to the email address to be delivered into a different mailbox or be discarded.

3. LIST command with the POSTADDRESS parameter

This document defines a new return option POSTADDRESS to the extended LIST command [\[LISTEXT\]](#) that requests the server to return an email address that can be used to post email to a mailbox returned by the LIST command. The POSTADDRESS return option causes the server to

Melnikov

Expires March 26, 2011

[Page 4]

Internet-Draft

IMAP4 POSTADDRESS extension

September 2010

return the LIST response with the POSTADDRESS information (see [section 4](#)).

The returned email address (if any, see below) doesn't have to be permanently available and MAY be different for different invocations of the LIST command. However if the returned address is generated anew each time, it MUST be valid for use for at least 2 hours since the moment it was generated.

If posting to the mailbox is not allowed or not supported the server MUST return NIL. For example, if the server also supports [\[ACL\]](#) extension and if the user that is issuing LIST RETURN (POSTADDRESS) is not granted the "p" right on the mailbox (the "p" right might be granted to the user directly, or through one of the groups the user belongs to, e.g. it may be granted to the "anonymous"), the extended LIST response MUST return NIL in POSTADDRESS information. Note, that the last requirement doesn't eliminate the need for the SMTP server to enforce access controls on delivery, as the returned email address may be passed by the IMAP client to a third party, not trusted by the SMTP server.

Also note, that if the server also supports [\[ACL\]](#) extension and if

the user doesn't have either "l" or "r" right on the mailbox, the server MUST NOT disclose the mailbox existence.

```
Example:  C: A002 LIST () "" INBOX RETURN (POSTADDRESS)
          S: * LIST () "/" INBOX ("POSTADDRESS" (
            "user1@example.com"))
          S: A002 OK List with postaddress info completed
```

Note that the empty () after the LIST command name are not required, which is shown below:

```
Example:  C: A002 LIST "" Drafts RETURN (POSTADDRESS)
          S: * LIST (\Marked) "/" Drafts ("POSTADDRESS" NIL)
          S: A002 OK List with postaddress info completed
```

The following 2 examples demonstrate email addresses that require [RFC 2821](#) quoting of the localpart:

```
Example:  C: A002 LIST "" "foo bar" RETURN (POSTADDRESS)
          S: * LIST () "/" "foo bar" ("POSTADDRESS" (
            "\"user1+foo bar\"@example.com"))
          S: A002 OK List with postaddress info completed
```

```
Example:  C: A002 LIST () "" "foo bar" RETURN (POSTADDRESS)
          S: * LIST () "/" "foo bar" (POSTADDRESS ({27}
            "user1+foo bar"@example.com))
          S: A002 OK List with postaddress info completed
```

The following example demonstrates that a non-existent subscribed mailbox doesn't have a corresponding post address:

```
Example:  C: A03 LIST (SUBSCRIBED) "" "*" RETURN (POSTADDRESS)
          ...
          S: * LIST (\Subscribed \NonExistent) "/" "Fruit/Peach"
            (POSTADDRESS NIL)
          ...
```

The SUBSCRIBED selection option is described in [[LISTEXT](#)].

[4.](#) Extended LIST response with POSTADDRESS information

Contents: name attributes
 hierarchy delimiter
 mailbox name
 email address for posting to the mailbox

This version of the LIST response occurs as a result of a LIST RETURN (POSTADDRESS) command. The proposed syntax conforms to the syntax of an extended LIST response as defined by mailbox-list ABNF element from [[LISTEXT](#)].

The meaning of "name attributes" and "hierarchy delimiter" is described in [section 7.2.2 of \[RFC3501\]](#). This is followed by the extension part that includes "POSTADDRESS" tag followed by an email address (enclosed in parenthesis) that can be used to post email to the mailbox. The returned email address MUST match the "Mailbox" ABNF production from [[RFC5321](#)]. If no such address exists for the mailbox, the server MUST return NIL.

The POSTADDRESS extended data item can occur only once in an extended LIST response. If the server knows multiple email addresses associated with a mailbox, it must return only one of them.

Example: S: * LIST () "/" Sent ("POSTADDRESS" (
 "user+Sent@example.com"))

[5.](#) Formal Syntax

Formal syntax is defined using ABNF [[ABNF](#)], extending the ABNF rules in [section 9 of \[RFC3501\]](#). Non-terminals referenced but not defined below are as defined in [[RFC3501](#)] and [[LISTEXT](#)].

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      =/ "POSTADDRESS"  
                ;;capability is defined in [RFC3501]
```

```
postaddr-label  = "POSTADDRESS"
```

```
return-option   =/ postaddr-label  
                ;; <return-option> is defined in [LISTTEXT]
```

```
postaddr-labret = postaddr-label /  
                  DQUOTE postaddr-label DQUOTE /  
                  "{11}" CRLF postaddr-label  
                  ;; POSTADDRESS label represented as IMAP atom,  
                  ;; quoted or literal string
```

```
postaddr-data   = postaddr-labret SP emaddr-or-nil  
                ;; postaddr-data conforms to the syntax of  
                ;; mbox-list-extended-item from [LISTTEXT]
```

```
emaddr-or-nil   = "(" email-address ")" /  
                NIL  
                ;; NIL if email address is not known
```

```
email-address   = astring
```

[6.](#) Security Considerations

Unless proper access restrictions are implemented, the POSTADDRESS extension can be used by a user to harvest email addresses. Note that email address harvesting is limited to users who already have

IMAP access to the service. Also note that some IMAP servers allow

for anonymous access.

Note that some implementations might return IMAP mailbox names in the addresses returned by POSTADDRESS (e.g. if "subaddressing" is used (see [Section 3.1.1 of \[RFC5598\]](#))), which might be considered a confidential information. Use of connection encryption such as TLS [\[RFC5246\]](#) is recommended to protect such confidential information.

Also note that interception of the addresses returned by the POSTADDRESS extension may enable a third party to inject mail to a specific mailbox. However this issue is not specific to this extension and can also be seen whenever "subaddressing" is used.

Additional security considerations are discussed in [Section 3](#).

[7](#). IANA Considerations

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

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

This document defines the POSTADDRESS IMAP capability. IANA is requested to add it to the registry.

IANA is also requested to register the following LISTEXT return option as specified in [\[LISTEXT\]](#):

To: iana@iana.org

Subject: Registration of LISTEXT option POSTADDRESS

LISTEXT option name: POSTADDRESS

LISTEXT option type: RETURN

LISTEXT option description: Causes the LIST command to return email address (if any) for posting to a returned mailbox.

Published specification : this RFC, [section 3](#).

Security considerations: this RFC, [section 6](#).

Intended usage: COMMON

Person & email address to contact for further information:

Alexey Melnikov <Alexey.Melnikov@isode.com>

Owner/Change controller: IESG <iesg@ietf.org>

8. Acknowledgements

The author would like to thank Ken Murchison for reminding that POSTADDRESS extension should not be a part of ACL2.

The author would also like to thank Dave Cridland, Philip Guenther, Arnt Gulbrandsen and Lisa Dusseault for corrections and suggestions to this document.

9. References

9.1. Normative References

- [ABNF] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", [RFC 5234](#), January 2008.
- [ACL] Melnikov, A., "IMAP4 Access Control List (ACL) Extension", [RFC 4314](#).
- [KEYWORDS] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), March 1997.
- [LISTEXT] Leiba, B. and A. Melnikov, "IMAP4 LIST Command Extensions", [RFC 5258](#), June 2008.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), March 2003.
- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", [RFC 5321](#), October 2008.

9.2. Informative References

- [RFC3030] Vaudreuil, G., "SMTP Service Extensions for Transmission of Large and Binary MIME Messages", [RFC 3030](#), December 2000.
- [RFC4467] Crispin, M., "Internet Message Access Protocol (IMAP) - URLAUTH Extension", [RFC 4467](#), May 2006.
- [RFC4468] Newman, C., "Message Submission BURL Extension", [RFC 4468](#), May 2006.
- [RFC5228] Guenther, P. and T. Showalter, "Sieve: An Email Filtering

Language", [RFC 5228](#), January 2008.

Melnikov

Expires March 26, 2011

[Page 9]

Internet-Draft

IMAP4 POSTADDRESS extension

September 2010

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", [RFC 5246](#), August 2008.

[RFC5598] Crocker, D., "Internet Mail Architecture", [RFC 5598](#), July 2009.

Author's Address

Alexey Melnikov
Isode Limited
5 Castle Business Village
36 Station Road
Hampton, Middlesex TW12 2BX
UK

Email: Alexey.Melnikov@isode.com

URI: <http://www.melnikov.ca/>

Melnikov

Expires March 26, 2011

[Page 10]