

Internet Engineering Task Force  
Internet-Draft  
Obsoletes: [RFC5738](#) (if approved)  
Intended status: Standards Track  
Expires: January 17, 2013

P. Resnick, Ed.  
Qualcomm Incorporated  
C. Newman, Ed.  
Oracle  
S. Shen, Ed.  
CNNIC  
July 16, 2012

**IMAP Support for UTF-8**  
**draft-ietf-eai-5738bis-06**

Abstract

This specification extends the Internet Message Access Protocol version 4rev1 (IMAP4rev1) to support UTF-8 encoded international characters in user names, mail addresses and message headers. This specification replaces [RFC 5738](#).

Status of This Memo

This Internet-Draft is submitted to IETF 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 January 17, 2013.

Copyright Notice

Copyright (c) 2012 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

<a href="#">1.</a>	Introduction . . . . .	<a href="#">3</a>
<a href="#">2.</a>	Conventions Used in this Document . . . . .	<a href="#">3</a>
<a href="#">3.</a>	UTF8=ACCEPT IMAP Capability . . . . .	<a href="#">3</a>
<a href="#">3.1.</a>	UTF-8 in IMAP Quoted Strings . . . . .	<a href="#">4</a>
<a href="#">4.</a>	IMAP UTF8 Append Data Extension . . . . .	<a href="#">5</a>
<a href="#">5.</a>	LOGIN Command and UTF-8 . . . . .	<a href="#">5</a>
<a href="#">6.</a>	UTF8=ONLY Capability . . . . .	<a href="#">6</a>
<a href="#">7.</a>	Dealing With Legacy Clients . . . . .	<a href="#">6</a>
<a href="#">8.</a>	Issues with UTF-8 Header Mailstore . . . . .	<a href="#">7</a>
<a href="#">9.</a>	IANA Considerations . . . . .	<a href="#">7</a>
<a href="#">10.</a>	Security Considerations . . . . .	<a href="#">7</a>
<a href="#">11.</a>	References . . . . .	<a href="#">8</a>
<a href="#">11.1.</a>	Normative References . . . . .	<a href="#">8</a>
<a href="#">11.2.</a>	Informative References . . . . .	<a href="#">9</a>
<a href="#">Appendix A.</a>	<a href="#">Appendix A.</a> Design Rationale . . . . .	<a href="#">9</a>
<a href="#">Appendix B.</a>	<a href="#">Appendix B.</a> Acknowledgments . . . . .	<a href="#">10</a>



## 1. Introduction

This specification extends IMAP4rev1 [RFC3501] to permit UTF-8 [RFC3629] in headers as described in "Internationalized Email Headers" [RFC6532]. It also adds a mechanism to support mailbox names using the UTF-8 charset. This specification creates two new IMAP capabilities to allow servers to advertise these new extensions.

Most of this specification assumes that the IMAP server will be operating in a fully internationalized environment, i.e., one in which all clients accessing the server will be able to accept non-ASCII message header fields and other information as specified in [Section 3](#). At least during a transition period, that assumption will not be realistic for many environments; the issues involved are discussed in [Section 7](#) below.

This specification replaces an earlier, experimental, approach to the same problem [RFC5738].

## 2. Conventions Used in this Document

The key words "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY" in this document are to be interpreted as defined in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

The formal syntax uses the Augmented Backus-Naur Form (ABNF) [RFC5234] notation. In addition, rules from IMAP4rev1 [RFC3501], UTF-8 [RFC3629], "Collected Extensions to IMAP4 ABNF" [RFC4466], and IMAP4 LIST Command Extensions [RFC5258] are also referenced.

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. UTF8=ACCEPT IMAP Capability

The "UTF8=ACCEPT" capability indicates that the server supports the ability to open mailboxes containing internationalized messages with SELECT and EXAMINE, and UTF-8 responses from the LIST and LSUB commands.

A client MUST use the "ENABLE UTF8=ACCEPT" command (defined in [RFC5161]) to indicate to the server that the client accepts UTF-8 in quoted-strings. The "ENABLE UTF8=ACCEPT" command MUST only be used in the authenticated state. (Note that the "UTF8=ONLY" capability described in [Section 6](#) imply the "UTF8=ACCEPT" capability. See



additional information in these sections.)

### 3.1. UTF-8 in IMAP Quoted Strings

The IMAP4rev1 [[RFC3501](#)] base specification forbids the use of 8-bit characters in atoms or quoted strings. Thus, a UTF-8 string can only be sent as a literal. This can be inconvenient from a coding standpoint, and unless the server offers IMAP4 non-synchronizing literals [[RFC2088](#)], this requires an extra round trip for each UTF-8 string sent by the client. When the IMAP server advertises the "UTF8=ACCEPT" capability, it informs the client that it supports UTF-8 in quoted-strings with the following syntax:

```
quoted          =/ DQUOTE *uQUOTED-CHAR DQUOTE
                  ; QUOTED-CHAR is not modified, as it will affect
                  ; other RFC 3501 ABNF non terminal.

uQUOTED-CHAR    = QUOTED-CHAR / UTF8-2 / UTF8-3 / UTF8-4

UTF8-2          = <Defined in Section 4 of RFC3629>

UTF8-3          = <Defined in Section 4 of RFC3629>

UTF8-4          = <Defined in Section 4 of RFC3629>
```

When this extended quoting mechanism is used by the client, then the server MUST reject octet sequences with the high bit set that fail to comply with the formal syntax in [[RFC3629](#)] with a BAD response. The IMAP server MUST NOT send UTF-8 in quoted strings to the client unless the client has indicated support for that syntax by using the "ENABLE UTF8=ACCEPT" command.

If the server advertises the "UTF8=ACCEPT" capability, the client MAY use extended quoted syntax with any IMAP argument that permits a string (including astring and nstring). However, if characters outside the US-ASCII repertoire are used in an inappropriate place, the results would be the same as if other syntactically valid but semantically invalid characters were used. Specific cases where UTF-8 characters are permitted or not permitted are described in the following paragraphs.

All IMAP servers that advertise the "UTF8=ACCEPT" capability SHOULD accept UTF-8 in mailbox names, and those that also support the "Mailbox International Naming Convention" described in [RFC 3501, Section 5.1.3](#) MUST accept utf8-quoted mailbox names and convert them to the appropriate internal format. Mailbox names MUST comply with the Net-Unicode Definition ([Section 2 of \[RFC5198\]](#)) with the specific exception that they MUST NOT contain control characters (0000-001F,



0080-009F), delete (007F), line separator (2028), or paragraph separator (2029).

An IMAP client MUST NOT issue a SEARCH command that uses a mixture of UTF-8 in quoted strings and a SEARCH CHARSET other than UTF-8. If an IMAP server receives such a SEARCH command, it SHOULD reject the command with a BAD response (due to the conflicting charset labels).

#### 4. IMAP UTF8 Append Data Extension

If the "UTF8=ACCEPT" capability is advertised, then the server accepts UTF-8 headers in the APPEND command message argument. A client that sends a message with UTF-8 headers to the server MUST send them using the "UTF8" APPEND data extension. If the server also advertises the CATENATE capability (as specified in [\[RFC4469\]](#)), the client can use the same data extension to include such a message in a CATENATE message part. The ABNF for the APPEND data extension and CATENATE extension follows:

```
utf8-literal    = "UTF8" SP "(" literal8 ")"
```

```
literal8       = <Defined in RFC 4466>
```

```
append-data     =/ utf8-literal
```

```
cat-part        =/ utf8-literal
```

IMAP servers that advertise support for "UTF8=ACCEPT" or "UTF8=ONLY" MUST reject an APPEND command that includes any 8-bit in the message headers with a "NO" response, when IMAP clients do not issue "ENABLE UTF8=ACCEPT" or "ENABLE UTF8=ONLY".

Note that the "UTF8=ONLY" capability described in [Section 6](#) implies the "UTF8=ACCEPT" capability. See additional information in that section.

#### 5. LOGIN Command and UTF-8

This specification doesn't extend the IMAP LOGIN command [\[RFC3501\]](#) to support UTF-8 usernames and passwords. Whenever a client needs to use UTF-8 username/passwords, it MUST use the IMAP AUTHENTICATE command which is already capable of passing UTF-8 user names and credentials.

Although this makes it syntactically legal to have a UTF-8 user name or password, there is no guarantee the user provisioning system used by the IMAP server will allow such identities. This is an implementation decision and MAY depend on what identity system the





IMAP server is configured to use.

## 6. UTF8=ONLY Capability

The "UTF8=ONLY" capability permits an IMAP server to advertise that it does not support the international mailbox name convention (modified UTF-7). As this is an incompatible change to IMAP, a clear warning is necessary. IMAP clients that find implementation of the "UTF8=ONLY" capability problematic are encouraged to at least detect the "UTF8=ONLY" capability and provide an informative error message to the end-user.

The "UTF8=ONLY" capability implies the "UTF8=ACCEPT" capability. UTF8=ACCEPT and UTF8=ONLY SHOULD be mutually exclusive. An IMAP server can advertise one of them, but never both.

## 7. Dealing With Legacy Clients

In most situations, it will be difficult or impossible for the implementer or operator of an IMAP (or POP) server to know whether all of the clients that might access it, or the associated mail store more generally, will be able to support the facilities defined in this document. In almost all cases, servers who conform to this specification will have to be prepared to deal with clients that do not enable the relevant capabilities. Unfortunately, there is no completely satisfactory way to do so other than for systems that wish to receive email that requires SMTPUTF8 capabilities to be sure that all components of those systems -- including IMAP and other clients selected by users -- are upgraded appropriately.

Choices available to the server when a message that requires SMTPUTF8 is encountered and the client doesn't enable UTF-8 capability include hiding the problematic message(s) as outlined elsewhere in this specification, creating in band or out of band notifications or error messages, or somehow trying to create a variation on the message with the intention of providing useful information to that client about what has occurred. Such variant messages cannot be actual substitutes for the original message: it will rarely be possible to reply to (either at all or without loss of information), new header fields or specialized constructs for server-client communication may go beyond the requirements of, e.g., [RFC 5322](#) and may consequently confuse some legacy mail user agents (including IMAP clients) or otherwise not provide the expected information to users. There are also tradeoffs in constructing variants of the original message between accepting complexity and additional computation costs in order to try to preserve as much information as possible (for example, in [\[popimap-downgrade\]](#)) and trying to minimize those costs while still providing useful information (for example, in



[[I-D.ietf-eai-simplifiedupgrade](#)]).

Because such messages are really variations on the original ones, not really "downgraded" (ones although that terminology is often used for convenience), they inevitably have relationships to the original ones that the IMAP specification [[RFC3501](#)] did not anticipate. In particular, digital signatures computed over the original message will often not be applicable to the variant version and servers that may be accessed by the same user with different clients or methods (e.g., POP or webmail systems in addition to IMAP or IMAP clients with different capabilities) will need to exert extreme care to be sure that UIDVALIDITY behaves as the user would expect. Those issues may be especially sensitive if the server caches the variant message or computes and stores it when the message arrives with the intent of making either form available depending on client capabilities.

The best (or "least bad") approach for any given environment will depend on local conditions, local assumptions about user behavior, the degree of control the server operator has over client usage and upgrading, the options that are actually available, and so on. It is impossible, at least at the time, to give good advice that will apply to all situations, or even particular profiles of situations, other than "upgrade legacy clients as soon as possible".

## **8. Issues with UTF-8 Header Mailstore**

When an IMAP server uses a mailbox format that supports UTF-8 headers and it permits selection or examination of that mailbox without the "UTF8" parameter, it is the responsibility of the server to comply with the IMAP4rev1 base specification [[RFC3501](#)] and [[RFC5322](#)] with respect to all header information transmitted over the wire. Mechanisms for 7-bit downgrading to help comply with the standards are discussed in [[popimap-downgrade](#)].

## **9. IANA Considerations**

This document adds two new capabilities ("UTF8=ACCEPT" and "UTF8=ONLY") to the IMAP4rev1 Capabilities registry [[RFC3501](#)]. Three other IMAP capabilities that were described in the experimental predecessor to this document (UTF8=ALL, UTF8=APPEND, UTF8=USER) are to be marked OBSOLETE in the registry.

## **10. Security Considerations**

The security considerations of UTF-8 [[RFC3629](#)] and SASLprep [[RFC4013](#)] apply to this specification, particularly with respect to use of UTF-8 in user names and passwords. Otherwise, this is not believed to alter the security considerations of IMAP4rev1.



Special considerations, some of them with security implications, occur if a server that conforms to this specification is accessed by a client that does not and in some more complex situations in which a given message is accessed by multiple clients that might use different protocols and/or support different capabilities. Those issues are discussed in [Section 7](#) above.

## **[11](#). References**

### **[11.1](#). Normative References**

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", [RFC 5198](#), March 2008.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", [RFC 3501](#), March 2003.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, [RFC 3629](#), November 2003.
- [RFC4013] Zeilenga, K., "SASLprep: Stringprep Profile for User Names and Passwords", [RFC 4013](#), February 2005.
- [RFC4466] Melnikov, A. and C. Daboo, "Collected Extensions to IMAP4 ABNF", [RFC 4466](#), April 2006.
- [RFC4469] Resnick, P., "Internet Message Access Protocol (IMAP) CATENATE Extension", [RFC 4469](#), April 2006.
- [RFC5161] Gulbrandsen, A. and A. Melnikov, "The IMAP ENABLE Extension", [RFC 5161](#), March 2008.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.



- [RFC5258] Leiba, B. and A. Melnikov, "Internet Message Access Protocol version 4 - LIST Command Extensions", [RFC 5258](#), June 2008.
- [RFC6532] Yang, A., Steele, S., and N. Freed, "Internationalized Email Headers", [RFC 6532](#), February 2012.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", [RFC 5322](#), October 2008.

### **11.2. Informative References**

- [RFC2088] Myers, J., "IMAP4 non-synchronizing literals", [RFC 2088](#), January 1997.
- [RFC5738] Resnick, P. and C. Newman, "IMAP Support for UTF-8", [RFC 5738](#), March 2010.
- [I-D.ietf-eai-simplesdowngrade] Gulbrandsen, A., "EAI: Simplified POP/IMAP downgrading", [draft-ietf-eai-simplesdowngrade-05](#) (work in progress), June 2012.
- [popimap-downgrade] Fujiwara, K., "Post-delivery Message Downgrading for Internationalized Email Messages", [draft-ietf-eai-popimap-downgrade-06](#) (work in progress), July 2012.

### **Appendix A. Appendix A. Design Rationale**

This non-normative section discusses the reasons behind some of the design choices in the above specification.

The basic approach of advertising the ability to access a mailbox in UTF-8 mode is intended to permit graceful upgrade, including servers that support multiple mailbox formats. In particular, it would be undesirable to force conversion of an entire server mailstore to UTF-8 headers, so being able to phase-in support for new mailboxes and gradually migrate old mailboxes is permitted by this design.

The "UTF8=ONLY" mechanism simplifies diagnosis of interoperability problems when legacy support goes away. In the situation where backwards compatibility is broken anyway, just-send-UTF-8 IMAP has the advantage that it might work with some legacy clients. However,





the difficulty of diagnosing interoperability problems caused by a just-send-UTF-8 IMAP mechanism is the reason the "UTF8=ONLY" capability mechanism was chosen.

## [Appendix B.](#) [Appendix B.](#) Acknowledgments

The authors wish to thank the participants of the EAI working group for their contributions to this document with particular thanks to Harald Alvestrand, David Black, Randall Gellens, Arnt Gulbrandsen, Kari Hurtt, John Klensin, Xiaodong Lee, Charles Lindsey, Alexey Melnikov, Subramanian Moonesamy, Shawn Steele, Daniel Taharlev, and Joseph Yee for their specific contributions to the discussion.

### Authors' Addresses

Pete Resnick (editor)  
Qualcomm Incorporated  
5775 Morehouse Drive  
San Diego, CA 92121-1714  
US

Phone: +1 858 651 4478  
EMail: [presnick@qualcomm.com](mailto:presnick@qualcomm.com)

Chris Newman (editor)  
Oracle  
800 Royal Oaks  
Monrovia, CA 91016  
USA

Phone:  
EMail: [chris.newman@oracle.com](mailto:chris.newman@oracle.com)

Sean Shen (editor)  
CNNIC  
No.4 South 4th Zhongguancun Street  
Beijing, 100190  
China

Phone: +86 10-58813038  
EMail: [shenshuo@cnnic.cn](mailto:shenshuo@cnnic.cn)

