

Internet Engineering Task Force  
Internet-Draft  
Obsoletes: RFC5738 (if approved)  
Intended status: Standards Track  
Expires: January 12, 2012

P. Resnick, Ed.  
Qualcomm Incorporated  
C. Newman, Ed.  
Oracle  
S. Shen, Ed.  
CNNIC  
July 11, 2011

IMAP Support for UTF-8  
draft-ietf-eai-5738bis-01

## 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.

## 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 12, 2012.

## Copyright Notice

Copyright (c) 2011 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

1. Introduction . . . . .	3
2. Conventions Used in this Document . . . . .	3
3. UTF8=ACCEPT IMAP Capability . . . . .	3
3.1. IMAP UTF-8 Quoted Strings . . . . .	4
3.2. UTF8 Parameter to SELECT and EXAMINE . . . . .	5
3.3. UTF-8 LIST and LSUB Responses . . . . .	6
3.4. UTF-8 Interaction with IMAP4 LIST Command Extensions . . . . .	6
3.4.1. UTF8 and UTF8ONLY LIST Selection Options . . . . .	6
3.4.2. UTF8 LIST Return Option . . . . .	7
4. UTF8=APPEND Capability . . . . .	7
5. UTF8=USER Capability . . . . .	8
6. UTF8=ALL Capability . . . . .	8
7. UTF8=ONLY Capability . . . . .	8
8. Up-Conversion Server Requirements . . . . .	9
9. Issues with UTF-8 Header Mailstore . . . . .	10
10. IANA Considerations . . . . .	10
11. Security Considerations . . . . .	11
12. References . . . . .	12
12.1. Normative References . . . . .	12
12.2. Informative References . . . . .	13
Appendix A. Appendix A. Design Rationale . . . . .	14
Appendix B. Appendix B. Examples Demonstrating Relationships between UTF8= Capabilities . . . . .	15
Appendix C. Appendix C. Acknowledgments . . . . .	15

## 1. Introduction

This specification extends IMAP4rev1 [RFC3501] to permit UTF-8 [RFC3629] in headers as described in "Internationalized Email Headers" [I-D.ietf-eai-rfc5335bis] . It also adds a mechanism to support mailbox names, login names, and passwords using the UTF-8 charset. This specification creates five new IMAP capabilities to allow servers to advertise these new extensions, along with two new IMAP LIST selection options and a new IMAP LIST return option.

This specification permits implementation of an IMAP server that hides mailboxes with internationalized email messages from IMAP clients that do not support this extension. Implementation of "Post-delivery Message Downgrading for Internationalized Email Messages" [popimap-downgrade] is necessary for an IMAP server to make mailboxes with internationalized email messages visible to IMAP clients that do not support this extension.

## 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 including the core rules defined in Appendix B of [RFC5234]. 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 UTF-8 quoted strings, the "UTF8" parameter to 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 quoted-strings. The "ENABLE UTF8=ACCEPT" command MUST only be used in the authenticated state. (Note that the "UTF8=ONLY" capability described in Section 7 and the "UTF8=ALL" capability described in Section 6 imply the "UTF8=ACCEPT" capability. See additional

information in these sections.)

### 3.1. IMAP UTF-8 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 native UTF-8 quoted-strings with the following syntax:

```
string          =/ uQuoted
uQuoted         = "*" DQUOTE *uQUOTED-CHAR DQUOTE
                  ; referred as 'utf8-quote' in this document
DQUOTE          = <Defined in appendix B.1 of RFC 5234>
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 quoting mechanism is used by the client (specifically an octet sequence beginning with "\*" and ending with "), 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 utf8-quoted syntax 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 utf8-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. For example, if the client includes UTF-8 characters in the user or password arguments (and the server has not advertised "UTF8=USER"), the LOGIN command will fail as it would with any other invalid user name or password. 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 utf8-quoted syntax 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).

### 3.2. UTF8 Parameter to SELECT and EXAMINE

The "UTF8=ACCEPT" capability also indicates that the server supports the "UTF8" parameter to SELECT and EXAMINE. When a mailbox is selected with the "UTF8" parameter, it alters the behavior of all IMAP commands related to message sizes, message headers, and MIME body headers so they refer to the message with UTF-8 headers. If the mailstore is not UTF-8 header native and the SELECT or EXAMINE command with UTF-8 header modifier succeeds, then the server MUST return results as if the mailstore were UTF-8 header native with upconversion requirements as described in Section 8. The server MAY reject the SELECT or EXAMINE command with the [NOT-UTF-8] response code, unless the "UTF8=ALL" or "UTF8=ONLY" capability is advertised.

Servers MAY include mailboxes that can only be selected or examined if the "UTF8" parameter is provided. However, such mailboxes MUST NOT be included in the output of an unextended LIST, LSUB, or equivalent command. If a client attempts to SELECT or EXAMINE such mailboxes without the "UTF8" parameter, the server MUST reject the command with a [UTF-8-ONLY] response code. As a result, such mailboxes will not be accessible by IMAP clients written prior to this specification and are discouraged unless the server advertises "UTF8=ONLY" or the server implements IMAP4 LIST Command Extensions [RFC5258].

utf8-select-param = "UTF8" ;; Conforms to select-param from RFC 4466

C: a SELECT newmailbox (UTF8)

S: ...

S: a OK SELECT completed

C: b FETCH 1 (SIZE ENVELOPE BODY)

S: ... UTF-8 header native results

S: b OK FETCH completed

C: c EXAMINE legacymailbox (UTF8)

S: c NO [NOT-UTF-8] Mailbox does not support UTF-8 access

C: d SELECT funky-new-mailbox

S: d NO [UTF-8-ONLY] Mailbox requires UTF-8 client

### 3.3. UTF-8 LIST and LSUB Responses

After an IMAP client successfully issues an "ENABLE UTF8=ACCEPT" command, the server MUST NOT return in LIST results any mailbox names to the client following the IMAP4 Mailbox International Naming Convention. Instead, the server MUST return any mailbox names with characters outside the US-ASCII repertoire using utf8-quoted syntax. (The IMAP4 Mailbox International Naming Convention has proved problematic in the past, so the desire is to make this syntax obsolete as quickly as possible.)

### 3.4. UTF-8 Interaction with IMAP4 LIST Command Extensions

When an IMAP server advertises both the "UTF8=ACCEPT" capability and the "LIST-EXTENDED" [RFC5258] capability, the server MUST support the LIST extensions described in this section.

#### 3.4.1. UTF8 and UTF8ONLY LIST Selection Options

The "UTF8" LIST selection option tells the server to include mailboxes that only support UTF-8 headers in the output of the list

command. The "UTF8ONLY" LIST selection option tells the server to include all mailboxes that support UTF-8 headers and to exclude mailboxes that don't support UTF-8 headers. Note that "UTF8ONLY" implies "UTF8", so it is not necessary for the client to request both. Use of either selection option will also result in UTF-8 mailbox names in the result as described in Section 3.3 and implies the "UTF8" List return option described in Section 3.4.2.

#### 3.4.2. UTF8 LIST Return Option

If the client supplies the "UTF8" LIST return option, then the server MUST include either the "\NoUTF8" or the "\UTF8Only" mailbox attribute as appropriate. The "\NoUTF8" mailbox attribute indicates that an attempt to SELECT or EXAMINE that mailbox with the "UTF8" parameter will fail with a [NOT-UTF-8] response code. The "\UTF8Only" mailbox attribute indicates that an attempt to SELECT or EXAMINE that mailbox without the "UTF8" parameter will fail with a [UTF-8-ONLY] response code. Note that computing this information may be expensive on some server implementations, so this return option should not be used unless necessary.

The ABNF [RFC5234] for these LIST extensions follows:

```
List-select-independent-opt =/ "UTF8"
                               ; List-select-independent-opt is defined in RFC 5258 Section 6

list-select-base-opt         =/ "UTF8ONLY"
                               ; list-select-base-opt is defined in RFC 5258 Section 6

return-option                 =/ "UTF8"
                               ; return-option is defined in RFC 5258 Section 6

mbx-list-oflag                =/ "\NoUTF8" / "\UTF8Only"
                               ; mbx-list-oflag is defined in RFC 3501 Section 9

resp-text-code                =/ "NOT-UTF-8" / "UTF-8-ONLY"
                               ; resp-text-code is defined in RFC 3501 Section 9
```

#### 4. UTF8=APPEND Capability

If the "UTF8=APPEND" 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 ")"  
append-data     =/ utf8-literal  
cat-part        =/ utf8-literal
```

A server that advertises "UTF8=APPEND" MAY fail for \NotUTF8 mailboxes with a NOT-UTF-8 response code. If this command does not fail, it MAY follow the requirements of the IMAP base specification and [RFC5322] for message fetching. Mechanisms for 7-bit downgrading to help comply with the standards are discussed in [popimap-downgrade].

IMAP servers that do not advertise the "UTF8=APPEND" or "UTF8=ONLY" capability SHOULD reject an APPEND command that includes any 8-bit in the message headers with a "NO" response.

Note that the "UTF8=ONLY" capability described in Section 7 implies the "UTF8=APPEND" capability. See additional information in that section.

#### 5. UTF8=USER Capability

If the "UTF8=USER" capability is advertised, that indicates the server accepts UTF-8 user names and passwords and applies SASLprep [RFC4013] to both arguments of the LOGIN command. The server MUST reject UTF-8 that fails to comply with the formal syntax in RFC 3629 [RFC3629] or if it encounters Unicode characters listed in Section 2.3 of SASLprep RFC 4013 [RFC4013].

#### 6. UTF8=ALL Capability

The "UTF8=ALL" capability indicates all server mailboxes support UTF-8 headers. Specifically, SELECT and EXAMINE with the "UTF8" parameter will never fail with a [NOT-UTF-8] response code.

Note that the "UTF8=ONLY" capability described in Section 7 implies the "UTF8=ALL" capability. See additional information in that section.

Note that the "UTF8=ALL" capability implies the "UTF8=ACCEPT" capability.

#### 7. 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), and does not permit selection or examination of any



mailbox unless the "UTF8" parameter is provided. 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, the "UTF8=ALL" capability, and the "UTF8=APPEND" capability. A server that advertises "UTF8=ONLY" need not advertise the three implicit capabilities.

## 8. Up-Conversion Server Requirements

When an IMAP4 server uses a traditional mailbox format that includes 7-bit headers and it chooses to permit access to that mailbox with the "UTF8" parameter, it **MUST** support minimal up-conversion as described in this section.

The server **MUST** support up-conversion of the following address header-fields in the message header: From, Sender, To, CC, Bcc, Resent-From, Resent-Sender, Resent-To, Resent-CC, Resent-Bcc, and Reply-To. This up-conversion **MUST** include address domains encoded according to Internationalizing Domain Names in Applications (IDNA) [RFC5890], and MIME header encoding [RFC2047] of display-names and any [RFC5322] comments.

The following charsets **MUST** be supported for up-conversion of MIME header encoding [RFC2047]: UTF-8, US-ASCII, ISO-8859-1, ISO-8859-2, ISO-8859-3, ISO-8859-4, ISO-8859-5, ISO-8859-6, ISO-8859-7, ISO-8859-8, ISO-8859-9, ISO-8859-10, ISO-8859-14, and ISO-8859-15. If the server supports other charsets in IMAP SEARCH or IMAP CONVERT [RFC5259], it **SHOULD** also support those charsets in this conversion.

Up-conversion of MIME header encoding of the following headers **MUST** also be implemented: Subject, Date ([RFC5322] comments only), Comments, Keywords, and Content-Description.

Server implementations also **SHOULD** up-convert all MIME body headers [RFC2045], **SHOULD** up-convert or remove the deprecated (and misused) "name" parameter [RFC1341] on Content-Type, and **MUST** up-convert the Content-Disposition [RFC2183] "filename" parameter, except when any of these are contained within a multipart/signed MIME body part (see below). These parameters can be encoded using the standard MIME parameter encoding [RFC2231] mechanism, or via non-standard use of MIME header encoding [RFC2047] in quoted strings.

The IMAP server **MUST NOT** perform up-conversion of headers and content of multipart/signed, as well as Original-Recipient and Return-Path.

## 9. 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].

An IMAP server with a mailbox that supports UTF-8 headers MUST comply with the protocol requirements implicit from Section 8. However, the code necessary for such compliance need not be part of the IMAP server itself in this case. For example, the minimal required up-conversion could be performed when a message is inserted into the IMAP-accessible mailbox.

## 10. IANA Considerations

This document adds five new capabilities ("UTF8=ACCEPT", "UTF8=USER", "UTF8=APPEND", "UTF8=ALL", and "UTF8=ONLY") to the IMAP4rev1 Capabilities registry [RFC3501].

This document adds two new IMAP4 list selection options and one new IMAP4 list return option.

### 1. LIST-EXTENDED option name: UTF8

LIST-EXTENDED option type: SELECTION

Implied return options(s): UTF8

LIST-EXTENDED option description: Causes the LIST response to include mailboxes that mandate the UTF8 SELECT/EXAMINE parameter.

Published specification: RFC 5738bis, Section 3.4.1

Security considerations: RFC 5738bis, Section 11

Intended usage: COMMON

Person and email address to contact for further information: see the Authors' Addresses at the end of this specification

Owner/Change controller: iesg@ietf.org

### 2. LIST-EXTENDED option name: UTF8ONLY

LIST-EXTENDED option type: SELECTION

Implied return options(s): UTF8

LIST-EXTENDED option description: Causes the LIST response to include mailboxes that mandate the UTF8 SELECT/EXAMINE parameter and exclude mailboxes that do not support the UTF8 SELECT/EXAMINE parameter.

Published specification: RFC 5738bis, Section 3.4.1

Security considerations: RFC 5738bis, Section 11

Intended usage: COMMON

Person and email address to contact for further information: see the Authors' Addresses at the end of this specification

Owner/Change controller: iesg@ietf.org

### 3. LIST-EXTENDED option name: UTF8

LIST-EXTENDED option type: RETURN

Implied return options(s): none

LIST-EXTENDED option description: Causes the LIST response to include \NoUTF8 and \UTF8Only mailbox attributes.

Published specification: RFC 5738bis, Section 3.4.1

Security considerations: RFC 5738bis, Section 11

Intended usage: COMMON

Person and email address to contact for further information: see the Authors' Addresses at the end of this specification

Owner/Change controller: iesg@ietf.org

## 11. 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.

[\*\*]

This document does not address downgrading scenarios, the security issues are discussed in [popimap-downgrade]

## 12. References

### 12.1. Normative References

- [RFC1341] Borenstein, N. and N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies", RFC 1341, June 1992.
- [RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
- [RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, November 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2183] Troost, R., Dorner, S., and K. Moore, "Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field", RFC 2183, August 1997.
- [RFC2231] Freed, N. and K. Moore, "MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations", RFC 2231, November 1997.
- [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.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, 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.
- [RFC5259] Melnikov, A. and P. Coates, "Internet Message Access Protocol - CONVERT Extension", RFC 5259, July 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [I-D.ietf-eai-rfc5335bis] Yang, A., Steele, S., and N. Freed, "Internationalized Email Headers", draft-ietf-eai-rfc5335bis-11 (work in progress), July 2011.
- [RFC5738] Resnick, P. and C. Newman, "IMAP Support for UTF-8", RFC 5738, March 2010.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.

## 12.2. Informative References

- [RFC2049] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Five: Conformance Criteria and Examples",

RFC 2049, November 1996.

- [RFC2088] Myers, J., "IMAP4 non-synchronizing literals", RFC 2088, January 1997.
- [RFC2277] Alvestrand, H., "IETF Policy on Character Sets and Languages", BCP 18, RFC 2277, January 1998.
- [I-D.ietf-eai-rfc5721bis] Gellens, R., Newman, C., Yao, J., and K. Fujiwara, "POP3 Support for UTF-8", draft-ietf-eai-rfc5721bis-01 (work in progress), March 2011.
- [popimap-downgrade] Fujiwara, K., "Post-delivery Message Downgrading for Internationalized Email Messages", draft-ietf-eai-popimap-downgrade-00 (work in progress), October 2010.

#### 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.

"UTF8=USER" is optional because many identity systems are US-ASCII only, so it's helpful to inform the client up front that UTF-8 won't work.

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.

The up-conversion requirements are designed to balance the desire to deprecate and eventually eliminate complicated encodings (like MIME header encodings) without creating a significant deployment burden for servers. As IMAP4 servers already require a MIME parser, this

includes additional server up-conversion requirements not present in POP3 Support for UTF-8 [I-D.draft-ietf-eai-rfc5721bis].

The set of mandatory charsets comes from two sources: MIME requirements [RFC2049] and IETF Policy on Character Sets [RFC2277]. Including a requirement to up-convert widely deployed encoded ideographic charsets to UTF-8 would be reasonable for most scenarios, but may require unacceptable table sizes for some embedded devices. The open-ended recommendation to support widely deployed charsets avoids the political ramifications of attempting to list such charsets. The authors believe market forces, existing open-source software, and public conversion tables are sufficient to deploy the appropriate charsets.

#### Appendix B. Appendix B. Examples Demonstrating Relationships between UTF8= Capabilities

UTF8=ACCEPT UTF8=USER UTF8=APPEND

UTF8=ACCEPT UTF8=ALL

UTF8=ALL ; Note, same as above

UTF8=ACCEPT UTF8=USER UTF8=APPEND UTF8=ALL UTF8=ONLY

UTF8=USER UTF8=ONLY ; Note, same as above

#### Appendix C. Appendix C. 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 Hurtta, 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

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)





Internet Engineering Task Force  
Internet-Draft  
Obsoletes: RFC5738 (if approved)  
Intended status: Standards Track  
Expires: May 20, 2013

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

IMAP Support for UTF-8  
draft-ietf-eai-5738bis-12

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 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 20, 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.

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

## 1. Introduction

This specification forms part of the Email Address Internationalization protocols described in the Email Address Internationalization Framework document [RFC6530]. It 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.

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. This document assumes that the reader will have a reasonably good understanding of the RFCs above.

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 and UTF-8 in IMAP Quoted Strings

The "UTF8=ACCEPT" capability indicates that the server supports the ability to open mailboxes containing internationalized messages with SELECT and EXAMINE, and can provide UTF-8 responses to the LIST and LSUB commands. This capability also affects other IMAP extensions that can return mailbox names or their prefixes, such as NAMESPACE

[RFC2342] and ACL [RFC4314].

The "UTF8=ONLY" capability described in Section 6 implies the "UTF8=ACCEPT" capability. A server is said to "support UTF8=ACCEPT" if it advertises either "UTF8=ACCEPT" or "UTF8=ONLY".

A client MUST use the "ENABLE" command (defined in [RFC5161]) with the "UTF8=ACCEPT" option (defined in Section 4 below) to indicate to the server that the client accepts UTF-8 in quoted-strings and supports UTF8=ACCEPT extension. The "ENABLE UTF8=ACCEPT" command is only valid in the authenticated state.

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 supports "UTF8=ACCEPT" 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 with a "BAD" response any octet sequences with the high bit set that fail to comply with the formal syntax in [RFC3629]. 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 supports "UTF8=ACCEPT", 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 support "UTF8=ACCEPT" 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 ([RFC5198], Section 2) with the specific exception that they MUST NOT contain control characters (0000-001F, 0080-009F), delete (007F), line separator (2028), or paragraph separator (2029).

Once an IMAP client has enabled UTF-8 support with the "ENABLE UTF8=ACCEPT" command, it MUST NOT issue a SEARCH command that contains a CHARSET specification. If an IMAP server receives such a SEARCH command in that situation, it SHOULD reject the command with a BAD response (due to the conflicting charset labels).

#### 4. IMAP UTF8 Append Data Extension

If the server supports "UTF8=ACCEPT", 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
```

If an IMAP server supports "UTF8=ACCEPT" and the IMAP client has not issued the "ENABLE UTF8=ACCEPT" command, the server MUST reject with a "NO" response an APPEND command that includes any 8-bit character in message header fields.

#### 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 the use of the IMAP AUTHENTICATE command in this way 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 indicates that the server supports "UTF8=ACCEPT" (see Section 4), and also that it requires support for UTF-8 from clients. In particular, this means that it will send UTF-8 in quoted strings, and it will not accept the older international mailbox name convention (modified UTF-7). Because these are incompatible changes to IMAP, explicit server announcement and client confirmation is necessary: clients MUST use the "ENABLE UTF8=ACCEPT" command before using this server. A server that advertises "UTF8=ONLY" will reject with a "NO [CANNOT]" response any command that might require UTF-8 support and is not preceded by an "ENABLE UTF8=ACCEPT" command.

IMAP clients that find support for a server that announces "UTF8=ONLY" problematic are encouraged to at least detect the announcement and provide an informative error message to the end-user.

Because the "UTF8=ONLY" server capability includes support for "UTF8=ACCEPT", the capability string will include at most one of those and never both. For the client, "ENABLE UTF8=ACCEPT" is always used -- never "ENABLE UTF8=ONLY".

## 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), 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: they will almost always be impossible to reply to (either at all or without loss of information); the new header fields or specialized constructs for server-client communication may go beyond the requirements of, e.g., RFC 5322; they may consequently confuse some legacy mail user agents (including IMAP clients) or otherwise may 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 [I-D.ietf-eai-popimap-downgrade]) and trying to minimize those costs while still providing useful information (for example, in [I-D.ietf-eai-simpledowngrade]).

Implementations that choose to do downgrading SHOULD use one of the standardized algorithms, [I-D.ietf-eai-popimap-downgrade] or [I-D.ietf-eai-simpledowngrade]. Getting downgrade algorithms right, and minimizing the risk of operational problems and harm to the email system, is tricky and requires careful engineering. These two algorithms are well understood and carefully designed.

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. This brings up two concerns in particular: First, digital signatures computed over and intended for the original message will often not be applicable to the variant message, and will often fail signature verification. (It will be possible for some digital signatures to be verified, if they cover only parts of the original message that are not affected in the creation of the variant.) Second, 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. Additionally, in order to cope with the case when a server compliant with this extension returns the same UIDVALIDITY to both legacy and UTF8=ACCEPT-aware clients, a client upgraded from non UTF8=ACCEPT aware MUST discard its cache of messages downloaded from the server.

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 of publication of this specification, 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 issuing "ENABLE UTF8=ACCEPT" first, 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. The issue of handling messages containing non-ASCII characters in legacy environments is discussed in Section 7.

## 9. IANA Considerations

This document redefines two capabilities ("UTF8=ACCEPT" and "UTF8=ONLY") in the IMAP 4 Capabilities registry [RFC3501]. Three other capabilities that were described in the experimental predecessor to this document (UTF8=ALL, UTF8=APPEND, UTF8=USER) are now made OBSOLETE. IANA is asked to change the IMAP 4 Capabilities registry as follows:

OLD:

UTF8=ACCEPT	[RFC5738]
UTF8=ALL	[RFC5738]
UTF8=APPEND	[RFC5738]
UTF8=ONLY	[RFC5738]
UTF8=USER	[RFC5738]

NEW:

UTF8=ACCEPT	[[this RFC]]
UTF8=ALL	OBSOLETE (was [RFC5738])
UTF8=APPEND	OBSOLETE (was [RFC5738])
UTF8=ONLY	[[this RFC]]
UTF8=USER	OBSOLETE (was [RFC5738])

## 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, as well as 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.
- [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.

- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, 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.
- [RFC6530] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", RFC 6530, February 2012.
- [I-D.ietf-eai-popimap-downgrade] Fujiwara, K., "Post-delivery Message Downgrading for Internationalized Email Messages", draft-ietf-eai-popimap-downgrade-08 (work in progress), October 2012.
- [I-D.ietf-eai-simpledowngrade] Gulbrandsen, A., "Simplified POP/IMAP Downgrading for Internationalized Email", draft-ietf-eai-simpledowngrade-07 (work in progress), August 2012.

## 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.
- [RFC2342] Gahrns, M. and C. Newman, "IMAP4 Namespace", RFC 2342, May 1998.

[RFC4314]

Melnikov, A., "IMAP4 Access Control List (ACL) Extension", RFC 4314, December 2005.

#### 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. 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 Hurtta, 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@qti.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)



Email Address Internationalization  
(EAI)  
Internet-Draft  
Obsoletes: 4952, 5504, 5825  
(if approved)  
Intended status: Standards Track  
Expires: April 30, 2012

J. Klensin  
Y. Ko  
October 28, 2011

Overview and Framework for Internationalized Email  
draft-ietf-eai-frmwrk-4952bis-12

Abstract

Full use of electronic mail throughout the world requires that (subject to other constraints) people be able to use close variations on their own names (written correctly in their own languages and scripts) as mailbox names in email addresses. This document introduces a series of specifications that define mechanisms and protocol extensions needed to fully support internationalized email addresses. These changes include an SMTP extension and extension of email header syntax to accommodate UTF-8 data. The document set also includes discussion of key assumptions and issues in deploying fully internationalized email. This document is a replacement for RFC 4952; it reflects additional issues identified since that document was published.

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 April 30, 2012.

Copyright Notice

Copyright (c) 2011 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 . . . . .	4
2. Role of This Specification . . . . .	5
3. Problem Statement . . . . .	5
4. Terminology . . . . .	6
4.1. Mail User and Mail Transfer Agents . . . . .	6
4.2. Address Character Sets . . . . .	7
4.3. User Types . . . . .	7
4.4. Messages . . . . .	8
4.5. Mailing Lists . . . . .	8
4.6. Conventional Message and Internationalized Message . . . . .	8
4.7. Undeliverable Messages, Notification, and Delivery Receipts . . . . .	8
5. Overview of the Approach and Document Plan . . . . .	9
6. Review of Experimental Results . . . . .	10
7. Overview of Protocol Extensions and Changes . . . . .	10
7.1. SMTP Extension for Internationalized Email Address . . . . .	10
7.2. Transmission of Email Header Fields in UTF-8 Encoding . . . . .	11
7.3. SMTP Service Extension for DSNs . . . . .	12
8. Downgrading before and after SMTP Transactions . . . . .	12
8.1. Downgrading before or during Message Submission . . . . .	13
8.2. Downgrading or Other Processing After Final SMTP Delivery . . . . .	14
9. Downgrading in Transit . . . . .	15
10. User Interface and Configuration Issues . . . . .	15



10.1. Choices of Mailbox Names and Unicode Normalization . . . .	16
11. Additional Issues . . . . .	17
11.1. Impact on URIs and IRIs . . . . .	17
11.2. Use of Email Addresses as Identifiers . . . . .	17
11.3. Encoded Words, Signed Messages, and Downgrading . . . . .	18
11.4. Other Uses of Local Parts . . . . .	18
11.5. Non-Standard Encapsulation Formats . . . . .	19
12. Key Changes From the Experimental Protocols and Framework . .	19
13. IANA Considerations . . . . .	19
14. Security Considerations . . . . .	19
15. Acknowledgments . . . . .	21
16. References . . . . .	21
16.1. Normative References . . . . .	21
16.2. Informative References . . . . .	22
Appendix A. Change Log . . . . .	26
A.1. Changes between -00 and -01 . . . . .	26
A.2. Changes between -01 and -02 . . . . .	27
A.3. Changes between -02 and -03 . . . . .	28
A.4. Changes between -03 and -04 . . . . .	28
A.5. Changes between -04 and -05 . . . . .	29
A.6. Changes between -05 and -06 . . . . .	29
A.7. Changes between -06 and -07 . . . . .	29
A.8. Changes between -07 and -08 (after IETF Last Call) . . . .	29
A.9. Changes between -08 and -09 . . . . .	29
A.10. Changes between -09 and -10 . . . . .	29
A.11. Changes between -10 and -11 . . . . .	29

## 1. Introduction

Note in Draft and to RFC Editor: The keyword represented in this document by "UTF8SMTPbis" (and in the XML source by &EASMTpkeyword;) is a placeholder. The actual keyword will be assigned when the standards track SMTP extension in this series [RFC5336bis-SMTP] is approved for publication and should be substituted here. This paragraph should be treated as normative reference to that SMTP extension draft, creating a reference hold until it is approved by the IESG. The paragraph should be removed before RFC publication.

In order to use internationalized email addresses, we need to internationalize both the domain part and the local part of email addresses. The domain part of email addresses is already internationalized [RFC5890], while the local part is not. Without the extensions specified in this document, the mailbox name is restricted to a subset of 7-bit ASCII [RFC5321]. Though MIME [RFC2045] enables the transport of non-ASCII data, it does not provide a mechanism for internationalized email addresses. In RFC 2047 [RFC2047], MIME defines an encoding mechanism for some specific message header fields to accommodate non-ASCII data. However, it does not permit the use of email addresses that include non-ASCII characters. Without the extensions defined here, or some equivalent set, the only way to incorporate non-ASCII characters in any part of email addresses is to use RFC 2047 coding to embed them in what RFC 5322 [RFC5322] calls the "display name" (known as a "name phrase" or by other terms elsewhere) of the relevant header fields. Information coded into the display name is invisible in the message envelope and, for many purposes, is not part of the address at all.

This document is a replacement for RFC 4952 [RFC4952]; it reflects additional issues, shared terminology, and some architectural changes identified since that document was published. It obsoletes that document. The experimental descriptions of in-transit downgrading [RFC5504][RFC5825], are now irrelevant and no longer needed due to the changes discussed in Section 12. The RFC Editor is requested to move all three of those documents to Historic.

The pronouns "he" and "she" are used interchangeably to indicate a human of indeterminate gender.

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 BCP 14, RFC 2119 [RFC2119]. Although this document is Informational, those requirements are consistent with requirements specified in the Standards Track documents in this set as described in Section 5.

## 2. Role of This Specification

This document presents the overview and framework for an approach to the next stage of email internationalization. This new stage requires not only internationalization of addresses and header fields, but also associated transport and delivery models. A prior version of this specification, RFC 4952 [RFC4952], also provided an introduction to a series of experimental protocols [RFC5335] [RFC5336] [RFC5337] [RFC5504] [RFC5721] [RFC5738] [RFC5825]. This revised form provides overview and conceptual information for the standards-track successors of a subset of those protocols. Details of the documents and the relationships among them appear in Section 5 and a discussion of what was learned from the Experimental protocols and their implementations appears in Section 6.

Taken together, these specifications provide the details for a way to implement and support internationalized email. The document itself describes how the various elements of email internationalization fit together and the relationships among the primary specifications associated with message transport, header formats, and handling.

This document, and others that comprise the collection described above, assume a reasonable familiarity with the basic Internet electronic mail specifications and terminology [RFC5321][RFC5322] and the MIME [RFC2045] and 8BITMIME [RFC6152] ones as well. While not strictly required to implement this specification, a general familiarity with the terminology and functions of IDNA [RFC5890][RFC5891] [RFC5892][RFC5893] [RFC5894] are also assumed.

## 3. Problem Statement

Internationalizing Domain Names in Applications (IDNA) [RFC5890] permits internationalized domain names, but deployment has not yet reached most users. One of the reasons for this is that we do not yet have fully internationalized naming schemes. Domain names are just one of the various names and identifiers that are required to be internationalized. In many contexts, until more of those identifiers are internationalized, internationalized domain names alone have little value.

Email addresses are prime examples of why it is not good enough to just internationalize the domain name. As most observers have learned from experience, users strongly prefer email addresses that resemble names or initials to those involving seemingly meaningless strings of letters or numbers. Unless the entire email address can use familiar characters and formats, users will perceive email as being culturally unfriendly. If the names and initials used in email addresses can be expressed in the native languages and writing

systems of the users, the Internet will be perceived as more natural, especially by those whose native language is not written in a subset of a Roman-derived script.

Internationalization of email addresses is not merely a matter of changing the SMTP envelope; or of modifying the From, To, and Cc header fields; or of permitting upgraded Mail User Agents (MUAs) to decode a special coding and respond by displaying local characters. To be perceived as usable, the addresses must be internationalized and handled consistently in all of the contexts in which they occur. This requirement has far-reaching implications: collections of patches and workarounds are not adequate. Even if they were adequate, a workaround-based approach may result in an assortment of implementations with different sets of patches and workarounds having been applied with consequent user confusion about what is actually usable and supported. Instead, we need to build a fully internationalized email environment, focusing on permitting efficient communication among those who share a language and writing system. That, in turn, implies changes to the mail header environment to permit those header fields that are appropriately internationalized to utilize the full range of Unicode characters, an SMTP Extension to permit UTF-8 [RFC3629] [RFC5198] mail addressing and delivery of those extended header fields, support for internationalization of delivery and service notifications [RFC3461] [RFC3464], and (finally) a requirement for support of the 8BITMIME SMTP Extension [RFC6152] so that all of these can be transported through the mail system without having to overcome the limitation that header fields do not have content-transfer-encodings.

#### 4. Terminology

This document assumes a reasonable understanding of the protocols and terminology of the core email standards as documented in [RFC5321] and [RFC5322].

##### 4.1. Mail User and Mail Transfer Agents

Much of the description in this document depends on the abstractions of "Mail Transfer Agent" ("MTA") and "Mail User Agent" ("MUA"). However, it is important to understand that those terms and the underlying concepts postdate the design of the Internet's email architecture and the application of the "protocols on the wire" principle to it. That email architecture, as it has evolved, and that "on the wire" principle have prevented any strong and standardized distinctions about how MTAs and MUAs interact on a given origin or destination host (or even whether they are separate).

However, the term "final delivery MTA" is used in this document in a

fashion equivalent to the term "delivery system" or "final delivery system" of RFC 5321. This is the SMTP server that controls the format of the local parts of addresses and is permitted to inspect and interpret them. It receives messages from the network for delivery to mailboxes or for other local processing, including any forwarding or aliasing that changes envelope addresses, rather than relaying. From the perspective of the network, any local delivery arrangements such as saving to a message store, handoff to specific message delivery programs or agents, and mechanisms for retrieving messages are all "behind" the final delivery MTA and hence are not part of the SMTP transport or delivery process.

#### 4.2. Address Character Sets

In this document, an address is "all-ASCII", or just an "ASCII address", if every character in the address is in the ASCII character repertoire [ASCII]; an address is "non-ASCII", or an "i18n-address", if any character is not in the ASCII character repertoire. Such addresses MAY be restricted in other ways, but those restrictions are not relevant to this definition. The term "all-ASCII" is also applied to other protocol elements when the distinction is important, with "non-ASCII" or "internationalized" as its opposite.

The umbrella term to describe the email address internationalization specified by this document and its companion documents is "UTF8SMTPbis".

[[anchor3: Note in Draft: Keyword to be changed before publication.]]  
For example, an address permitted by this specification is referred to as a "UTF8SMTPbis (compliant) address".

Please note that, according to the definitions given here, the set of all "all-ASCII" addresses and the set of all "non-ASCII" addresses are mutually exclusive. The set of all addresses permitted when UTF8SMTPbis appears is the union of these two sets.

#### 4.3. User Types

An "ASCII user" (i) exclusively uses email addresses that contain ASCII characters only, and (ii) cannot generate recipient addresses that contain non-ASCII characters.

An "i18mail user" has one or more non-ASCII email addresses, or is able to generate recipient addresses that contain non-ASCII characters. Such a user may have ASCII addresses too; if the user has more than one email account and a corresponding address, or more than one alias for the same address, he or she has some method to choose which address to use on outgoing email. Note that under this definition, it is not possible to tell from an ASCII address if the

owner of that address is an il8mail user or not. (A non-ASCII address implies a belief that the owner of that address is an il8mail user.) There is no such thing as an "il8mail message"; the term applies only to users and their agents and capabilities. In particular, the use of non-ASCII message content is an integral part of the MIME specifications [RFC2045] and does not require these extensions (although it is compatible with them).

#### 4.4. Messages

A "message" is sent from one user (sender) using a particular email address to one or more other recipient email addresses (often referred to just as "users" or "recipient users").

#### 4.5. Mailing Lists

A "mailing list" is a mechanism whereby a message may be distributed to multiple recipients by sending it to one recipient address. An agent (typically not a human being) at that single address then causes the message to be redistributed to the target recipients. This agent sets the envelope return address of the redistributed message to a different address from that of the original single recipient message. Using a different envelope return address (reverse-path) causes error (and other automatically generated) messages to go to an error handling address.

Special provisions for managing mailing lists that might contain non-ASCII addresses are discussed in a document that is specific to that topic [RFC5983] [RFC5983bis-MailingList].

#### 4.6. Conventional Message and Internationalized Message

- o A conventional message is one that does not use any extension defined in the SMTP extension document [RFC5336] or in the UTF8header specification [RFC5335], and is strictly conformant to RFC 5322 [RFC5322].
- o An internationalized message is a message utilizing one or more of the extensions defined in this set of specifications, so that it is no longer conformant to the traditional specification of an email message or its transport.

#### 4.7. Undeliverable Messages, Notification, and Delivery Receipts

As specified in RFC 5321, a message that is undeliverable for some reason is expected to result in notification to the sender. This can occur in either of two ways. One, typically called "Rejection", occurs when an SMTP server returns a reply code indicating a fatal

error (a "5yz" code) or persistently returns a temporary failure error (a "4yz" code). The other involves accepting the message during SMTP processing and then generating a message to the sender, typically known as a "Non-delivery Notification" or "NDN". Current practice often favors rejection over NDNs because of the reduced likelihood that the generation of NDNs will be used as a spamming technique. The latter, NDN, case is unavoidable if an intermediate MTA accepts a message that is then rejected by the next-hop server.

A sender MAY also explicitly request message receipts [RFC3461] that raise the same issues for these internationalization extensions as NDNs.

## 5. Overview of the Approach and Document Plan

This set of specifications changes both SMTP and the character encoding of email message headers to permit non-ASCII characters to be represented directly. Each important component of the work is described in a separate document. The document set, whose members are described below, also contains informational documents whose purpose is to provide implementation suggestions and guidance for the protocols.

In addition to this document, the following documents make up this specification and provide advice and context for it.

- o SMTP extension. The SMTP extension document [RFC5336bis-SMTP] provides an SMTP extension (as provided for in RFC 5321) for internationalized addresses.
- o Email message headers in UTF-8. The email message header document [RFC5335bis-Hdrs] essentially updates RFC 5322 to permit some information in email message headers to be expressed directly by Unicode characters encoded in UTF-8 when the SMTP extension described above is used. This document, possibly with one or more supplemental ones, will also need to address the interactions with MIME, including relationships between UTF8SMTPbis and internal MIME headers and content types.
- o Extensions to delivery status and notification handling to adapt to internationalized addresses [RFC5337bis-DSN].
- o Forthcoming documents will specify extensions to the IMAP protocol [RFC3501] to support internationalized message headers [RFC5738bis-IMAP], Parallel extensions to the POP protocol [RFC5721] [RFC5721bis-POP3], and some common properties of the two [POPIMAP-downgrade].

## 6. Review of Experimental Results

The key difference between this set of protocols and the experimental set that preceded them [RFC5335] [RFC5336] [RFC5337] [RFC5504] [RFC5721] [RFC5738] [RFC5825] is that the earlier group provided a mechanism for in-transit downgrading of messages (described in detail in RFC 5504). That mechanism permitted, and essentially required, that each non-ASCII address be accompanied by an all-ASCII equivalent. That, in turn, raised security concerns associated with pairing of addresses that could not be authenticated. It also introduced the first incompatible change to Internet mail addressing in many years, raising concerns about interoperability issues if the new address forms "leaked" into legacy email implementations. The WG concluded that the advantages of in-transit downgrading, were it feasible operationally, would be significant enough to overcome those concerns.

That turned out not to be the case, with interoperability problems among initial implementations. Prior to starting on the work that led to this set of specifications, the WG concluded that the combination of requirements and long-term implications of that earlier model were too complex to be satisfactory and that work should move ahead without it.

The other significant change to the protocols themselves is that the UTF8SMTPbis keyword is now required as an SMTP client announcement if the extension is needed; in the experimental version, only the server announcement that an extended envelope and/or content were permitted was necessary.

## 7. Overview of Protocol Extensions and Changes

### 7.1. SMTP Extension for Internationalized Email Address

An SMTP extension, "UTF8SMTPbis" is specified as follows:

- o Permits the use of UTF-8 strings in email addresses, both local parts and domain names.
- o Permits the selective use of UTF-8 strings in email message headers (see Section 7.2).
- o Requires that the server advertise the 8BITMIME extension [RFC6152] and that the client support 8-bit transmission so that header information can be transmitted without using a special content-transfer-encoding.

Some general principles affect the development decisions underlying



this work.

1. Email addresses enter subsystems (such as a user interface) that may perform charset conversions or other encoding changes. When the local part of the address includes characters outside the ASCII character repertoire, use of ASCII-compatible encoding (ACE) [RFC3492] [RFC5890] in the domain part is discouraged to promote consistent processing of characters throughout the address.
2. An SMTP relay MUST
  - \* Either recognize the format explicitly, agreeing to do so via an ESMTP option, or
  - \* Reject the message or, if necessary, return a non-delivery notification message, so that the sender can make another plan.
3. If the message cannot be forwarded because the next-hop system cannot accept the extension, it MUST be rejected or a non-delivery message MUST be generated and sent.
4. In the interest of interoperability, charsets other than UTF-8 are prohibited in mail addresses and message headers being transmitted over the Internet. There is no practical way to identify multiple charsets properly with an extension similar to this without introducing great complexity.

Conformance to the group of standards specified here for email transport and delivery requires implementation of the SMTP Extension specification and the UTF-8 Header specification. If the system implements IMAP or POP, it MUST conform to the i18n IMAP [RFC5738bis-IMAP] or POP [RFC5721bis-POP3] specifications respectively.

## 7.2. Transmission of Email Header Fields in UTF-8 Encoding

There are many places in MUAs or in a user presentation in which email addresses or domain names appear. Examples include the conventional From, To, or Cc header fields; Message-ID and In-Reply-To header fields that normally contain domain names (but that may be a special case); and in message bodies. Each of these must be examined from an internationalization perspective. The user will expect to see mailbox and domain names in local characters, and to see them consistently. If non-obvious encodings, such as protocol-specific ASCII-Compatible Encoding (ACE) variants, are used, the user will inevitably, if only occasionally, see them rather than

"native" characters and will find that discomfiting or astonishing. Similarly, if different codings are used for mail transport and message bodies, the user is particularly likely to be surprised, if only as a consequence of the long-established "things leak" principle. The only practical way to avoid these sources of discomfort, in both the medium and the longer term, is to have the encodings used in transport be as similar to the encodings used in message headers and message bodies as possible.

When email local parts are internationalized, they SHOULD be accompanied by arrangements for the message headers to be in the fully internationalized form. That form SHOULD use UTF-8 rather than ASCII as the base character set for the contents of header fields (protocol elements such as the header field names themselves are unchanged and remain entirely in ASCII). For transition purposes and compatibility with legacy systems, this can be done by extending the traditional MIME encoding models for non-ASCII characters in headers [RFC2045] [RFC2231], but even these should be based on UTF-8, rather than other encodings, if at all possible [RFC6055]. However, the target is fully internationalized message headers, as discussed in [RFC5335bis-Hdrs] and not an extended and painful transition.

### 7.3. SMTP Service Extension for DSNs

The existing Draft Standard Delivery status notifications (DSNs) specification [RFC3461] is limited to ASCII text in the machine readable portions of the protocol. "International Delivery and Disposition Notifications" [RFC5337bis-DSN] adds a new address type for international email addresses so an original recipient address with non-ASCII characters can be correctly preserved even after downgrading. If an SMTP server advertises both the UTF8SMTPbis and the DSN extension, that server MUST implement internationalized DSNs including support for the ORCPT parameter specified in RFC 3461 [RFC3461].

### 8. Downgrading before and after SMTP Transactions

An important issue with these extensions is how to handle interactions between systems that support non-ASCII addresses and legacy systems that expect ASCII. There is, of course, no problem with ASCII-only systems sending to those that can handle internationalized forms because the ASCII forms are just a proper subset. But, when systems that support these extensions send mail, they MAY include non-ASCII addresses for senders, receivers, or both and might also provide non-ASCII header information other than addresses. If the extension is not supported by the first-hop system (SMTP server accessed by the Submission server acting as an SMTP client), message originating systems SHOULD be prepared to either

send conventional envelopes and message headers or to return the message to the originating user so the message may be manually downgraded to the traditional form, possibly using encoded words [RFC2047] in the message headers. Of course, such transformations imply that the originating user or system must have ASCII-only addresses available for all senders and recipients. Mechanisms by which such addresses may be found or identified are outside the scope of these specifications as are decisions about the design of originating systems such as whether any required transformations are made by the user, the originating MUA, or the Submission server.

A somewhat more complex situation arises when the first-hop system supports these extensions but some subsequent server in the SMTP transmission chain does not. It is important to note that most cases of that situation with forward-pointing addresses will be the result of configuration errors: especially if it hosts non-ASCII addresses, a final delivery MTA that accepts these extensions SHOULD NOT be configured with lower-preference MX hosts that do not. When the only non-ASCII address being transmitted is backward-pointing (e.g., in an SMTP MAIL command), recipient configuration can not help in general. On the other hand, alternate, all-ASCII, addresses for senders are those most likely to be authoritatively known by the submission environment or the sender herself. Consequently, if an intermediate SMTP relay that requires these extensions then discovers that the next system in the chain does not support them, it will have little choice other than to reject or return the message.

As discussed above, downgrading to an ASCII-only form may occur before or during the initial message submission. It might also occur after the delivery to the final delivery MTA in order to accommodate messages stores or IMAP or POP servers or clients that have different capabilities than the delivery MTA. These two cases are discussed in the subsections below.

#### 8.1. Downgrading before or during Message Submission

The IETF has traditionally avoided specifying the precise behavior of MUAs to provide maximum flexibility in the associated user interfaces. The SMTP standard [RFC5321], Section 6.4, gives wide latitude to MUAs and Submission servers as to what might be supplied by the user as long as the result conforms with "on the wire" standards once it is injected into the public Internet. In that tradition, the discussion in the remainder of Section 8 is provided as general guidance rather than normative requirements.

Messages that require these extensions will sometimes be transferred to a system that does not support these extensions; it is likely that the most common cases will involve the combination of ASCII-only

forward-pointing addresses with a non-ASCII backward-pointing one. Until the extensions described here have been universally implemented in the Internet email environment, senders who prefer to use non-ASCII addresses (or raw UTF-8 characters in header fields) even when their intended recipients use and expect all-ASCII ones will need to be especially careful about the error conditions that can arise, especially if they are working in an environment in which non-delivery messages (or other indications from submission servers) are routinely dropped or ignored.

Perhaps obviously, the most convenient time to find an ASCII address corresponding to an internationalized address is at the originating MUA or closely-associated systems. This can occur either before the message is sent or after the internationalized form of the message is rejected. It is also the most convenient time to convert a message from the internationalized form into conventional ASCII form or to generate a non-delivery message to the sender if either is necessary. At that point, the user has a full range of choices available, including changing backward-pointing addresses, contacting the intended recipient out of band for an alternate address, consulting appropriate directories, arranging for translation of both addresses and message content into a different language, and so on. While it is natural to think of message downgrading as optimally being a fully-automated process, we should not underestimate the capabilities of a user of at least moderate intelligence who wishes to communicate with another such user.

In this context, one can easily imagine modifications to message submission servers (as described in RFC 4409 [RFC4409]) so that they would perform downgrading operations or perhaps even upgrading ones. Such operations would permit receiving messages with one or more of the internationalization extensions discussed here and adapting the outgoing message, as needed, to respond to the delivery or next-hop environment the submission server encounters.

## 8.2. Downgrading or Other Processing After Final SMTP Delivery

When an email message is received by a final delivery MTA, it is usually stored in some form. Then it is retrieved either by software that reads the stored form directly or by client software via some email retrieval mechanisms such as POP or IMAP.

The SMTP extension described in Section 7.1 provides protection only in transport. It does not prevent MUAs and email retrieval mechanisms that have not been upgraded to understand internationalized addresses and UTF-8 message headers from accessing stored internationalized emails.

Since the final delivery MTA (or, to be more specific, its corresponding mail storage agent) cannot safely assume that agents accessing email storage will always be capable of handling the extensions proposed here, it MAY downgrade internationalized emails, specially identify messages that utilize these extensions, or both. If this is done, the final delivery MTA SHOULD include a mechanism to preserve or recover the original internationalized forms without information loss to support access by UTF8SMTPbis-aware agents.

#### 9. Downgrading in Transit

The base SMTP specification (Section 2.3.11 of RFC 5321 [RFC5321]) states that "due to a long history of problems when intermediate hosts have attempted to optimize transport by modifying them, the local-part MUST be interpreted and assigned semantics only by the host specified in the domain part of the address". This is not a new requirement; equivalent statements appeared in specifications in 2001 [RFC2821] and even in 1989 [RFC1123].

Adherence to this rule means that a downgrade mechanism that transforms the local-part of an email address cannot be utilized in transit. It can only be applied at the endpoints, specifically by the MUA or submission server or by the final delivery MTA.

One of the reasons for this rule has to do with legacy email systems that embed mail routing information in the local-part of the address field. Transforming the email address destroys such routing information. There is no way a server other than the final delivery server can know, for example, whether the local-part of user%foo@example.com is a route ("user" is reached via "foo") or simply a local address.

#### 10. User Interface and Configuration Issues

Internationalization of addresses and message headers, especially in combination with variations on character coding that are inherent to Unicode, may make careful choices of addresses and careful configuration of servers and DNS records even more important than they are for traditional Internet email. It is likely that, as experience develops with the use of these protocols, it will be desirable to produce one or more additional documents that offer guidance for configuration and interfaces. A document that discusses issues with mail user agents (MUAs), especially with regard to downgrading, is expected to be developed in the EAI Working Group. The subsections below address some other issues.

### 10.1. Choices of Mailbox Names and Unicode Normalization

It has long been the case that the email syntax permits choices about mailbox names that are unwise in practice if one actually intends the mailboxes to be accessible to a broad range of senders. The most-often-cited examples involve the use of case-sensitivity and tricky quoting of embedded characters in mailbox local parts. These deliberately-unusual constructions are permitted by the protocols and servers are expected to support them. Although they can provide value in special cases, taking advantage of them is almost always bad practice unless the intent is to create some form of security by obscurity.

In the absence of these extensions, SMTP clients and servers are constrained to using only those addresses permitted by RFC 5321. The local parts of those addresses MAY be made up of any ASCII characters except the control characters that 5321 prohibits, although some of them MUST be quoted as specified there. It is notable in an internationalization context that there is a long history on some systems of using overstruck ASCII characters (a character, a backspace, and another character) within a quoted string to approximate non-ASCII characters. This form of internationalization was permitted by RFC 821 [RFC0821] but is prohibited by RFC 5321 because it requires a backspace character (a prohibited C0 control). Because RFC 5321 (and its predecessor, RFC 2821) prohibit the use of this character in ASCII mailbox names and it is even more problematic (for canonicalization and normalization reasons) in non-ASCII strings, backspace MUST NOT appear in UTF8SMTPbis mailbox names.

For the particular case of mailbox names that contain non-ASCII characters in the local part, domain part, or both, special attention MUST be paid to Unicode normalization [Unicode-UAX15], in part because Unicode strings may be normalized by other processes independent of what a mail protocol specifies (this is exactly analogous to what may happen with quoting and dequoting in traditional addresses). Consequently, the following principles are offered as advice to those who are selecting names for mailboxes:

- o In general, it is wise to support addresses in Normalized form, using at least Normalization Form NFC. Except in circumstances in which NFKC would map characters together that the parties responsible for the destination mail server would prefer to be kept distinguishable, supporting the NFKC-conformant form would yield even more predictable behavior for the typical user.
- o It will usually be wise to support other forms of the same local-part string, either as aliases or by normalization of strings reaching the delivery server: the sender should not be depended

upon to send the strings in normalized form.

- o Stated differently and in more specific terms, the rules of the protocol for local-part strings essentially provide that:
  - \* Unnormalized strings are valid, but sufficiently bad practice that they may not work reliably on a global basis. Servers should not depend on clients to send normalized forms but should be aware that procedures on client machines outside the control of the MUA may cause normalized strings to be sent regardless of user intent.
  - \* C0 (and presumably C1) controls (see The Unicode Standard [Unicode]) are prohibited, the first in RFC 5321 and the second by an obvious extension from it [RFC5198].
  - \* Other kinds of punctuation, spaces, etc., are risky practice. Perhaps they will work, and SMTP receiver code is required to handle them without severe errors (even if such strings are not accepted in addresses to be delivered on that server), but creating dependencies on them in mailbox names that are chosen is usually a bad practice and may lead to interoperability problems.

## 11. Additional Issues

This section identifies issues that are not covered, or not covered comprehensively, as part of this set of specifications, but that will require ongoing review as part of deployment of email address and header internationalization.

### 11.1. Impact on URIs and IRIs

The mailto: schema [RFC6068], and the discussion of it in the Internationalized Resource Identifier (IRI) specification [RFC3987] may need to be modified when this work is completed and standardized.

### 11.2. Use of Email Addresses as Identifiers

There are a number of places in contemporary Internet usage in which email addresses are used as identifiers for individuals, including as identifiers to Web servers supporting some electronic commerce sites and in some X.509 certificates [RFC5280]. These documents do not address those uses, but it is reasonable to expect that some difficulties will be encountered when internationalized addresses are first used in those contexts, many of which cannot even handle the full range of addresses permitted today.

### 11.3. Encoded Words, Signed Messages, and Downgrading

One particular characteristic of the email format is its persistency: MUAs are expected to handle messages that were originally sent decades ago and not just those delivered seconds ago. As such, MUAs and mail filtering software, such as that specified in Sieve [RFC5228], will need to continue to accept and decode header fields that use the "encoded word" mechanism [RFC2047] to accommodate non-ASCII characters in some header fields. While extensions to both POP3 [RFC1939] and IMAP [RFC3501] have been defined that include automatic upgrading of messages that carry non-ASCII information in encoded form -- including RFC 2047 decoding -- of messages by the POP3 [RFC5721bis-POP3] or IMAP [RFC5738bis-IMAP] server, there are message structures and MIME content-types for which that cannot be done or where the change would have unacceptable side effects.

For example, message parts that are cryptographically signed, using e.g., S/MIME [RFC5751] or Pretty Good Privacy (PGP) [RFC3156], cannot be upgraded from the RFC 2047 form to normal UTF-8 characters without breaking the signature. Similarly, message parts that are encrypted may contain, when decrypted, header fields that use the RFC 2047 encoding; such messages cannot be 'fully' upgraded without access to cryptographic keys.

Similar issues may arise if messages are signed and then subsequently downgraded, e.g., as discussed in Section 8.1, and then an attempt is made to upgrade them to the original form and then verify the signatures. Even the very subtle changes that may result from algorithms to downgrade and then upgrade again may be sufficient to invalidate the signatures if they impact either the primary or MIME bodypart headers. When signatures are present, downgrading **MUST** be performed with extreme care if at all.

### 11.4. Other Uses of Local Parts

Local parts are sometimes used to construct domain labels, e.g., the local part "user" in the address user@domain.example could be converted into a vanity host user.domain.example with its Web space at <http://user.domain.example> and the catchall addresses any.thing.goes@user.domain.example.

Such schemes are obviously limited by, among other things, the SMTP rules for domain names, and will not work without further restrictions for other local parts such as the <utf8-local-part> specified in [RFC5335bis-Hdrs]. Whether those limitations are relevant to these specifications is an open question. It may be simply another case of the considerable flexibility accorded to delivery MTAs in determining the mailbox names they will accept and



how they are interpreted.

#### 11.5. Non-Standard Encapsulation Formats

Some applications use formats similar to the application/mbox format defined in [RFC4155] instead of the message/digest form described in RFC 2046, Section 5.1.5 [RFC2046] to transfer multiple messages as single units. Insofar as such applications assume that all stored messages use the message/rfc822 format described in RFC 2046, Section 5.2.1 [RFC2046] with ASCII message headers, they are not ready for the extensions specified in this series of documents and special measures may be needed to properly detect and process them.

#### 12. Key Changes From the Experimental Protocols and Framework

The original framework for internationalized email addresses and headers was described in RFC 4952 and a subsequent set of experimental protocol documents. Those relationships are described in Section 3. The key architectural difference between the experimental specifications and this newer set is that the earlier specifications supported in-transit downgrading. Those mechanisms included the definition of syntax and functions to support passing alternate, all-ASCII, addresses with the non-ASCII ones as well as special headers to indicate the downgraded status of messages. Those features were eliminated after experimentation indicated that they were more complex and less necessary than had been assumed earlier. Those issues are described in more detail in Section 6 and Section 9.

#### 13. IANA Considerations

This overview description and framework document does not contemplate any IANA registrations or other actions. Some of the documents in the group have their own IANA considerations sections and requirements.

#### 14. Security Considerations

Any expansion of permitted characters and encoding forms in email addresses raises some risks. There have been discussions on so called "IDN-spoofing" or "IDN homograph attacks". These attacks allow an attacker (or "phisher") to spoof the domain or URLs of businesses. The same kind of attack is also possible on the local part of internationalized email addresses. It should be noted that the proposed fix involving forcing all displayed elements into normalized lower-case works for domain names in URLs, but not for email local parts since those are case sensitive.

Since email addresses are often transcribed from business cards and

notes on paper, they are subject to problems arising from confusable characters (see [RFC4690]). These problems are somewhat reduced if the domain associated with the mailbox is unambiguous and supports a relatively small number of mailboxes whose names follow local system conventions. They are increased with very large mail systems in which users can freely select their own addresses.

The internationalization of email addresses and message headers must not leave the Internet less secure than it is without the required extensions. The requirements and mechanisms documented in this set of specifications do not, in general, raise any new security issues.

They do require a review of issues associated with confusable characters -- a topic that is being explored thoroughly elsewhere (see, e.g., RFC 4690 [RFC4690]) -- and, potentially, some issues with UTF-8 normalization, discussed in RFC 3629 [RFC3629], and other transformations. Normalization and other issues associated with transformations and standard forms are also part of the subject of work described elsewhere [RFC5198] [RFC5893] [RFC6055].

Some issues specifically related to internationalized addresses and message headers are discussed in more detail in the other documents in this set. However, in particular, caution should be taken that any "downgrading" mechanism, or use of downgraded addresses, does not inappropriately assume authenticated bindings between the internationalized and ASCII addresses. This potential problem can be mitigated somewhat by enforcing the expectation that most or all such transformations will be performed prior to final delivery by systems that are presumed to be under the administrative control of the sending user (as opposed to being performed in transit by entities that are not under the administrative control of the sending user).

The new UTF-8 header and message formats might also raise, or aggravate, another known issue. If the model creates new forms of an 'invalid' or 'malformed' message, then a new email attack is created: in an effort to be robust, some or most agents will accept such message and interpret them as if they were well-formed. If a filter interprets such a message differently than the MUA used by the recipient, then it may be possible to create a message that appears acceptable under the filter's interpretation but that should be rejected under the interpretation given to it by that MUA. Such attacks already exist for existing messages and encoding layers, e.g., invalid MIME syntax, invalid HTML markup, and invalid coding of particular image types.

In addition, email addresses are used in many contexts other than sending mail, such as for identifiers under various circumstances (see Section 11.2). Each of those contexts will need to be

evaluated, in turn, to determine whether the use of non-ASCII forms is appropriate and what particular issues they raise.

This work will clearly affect any systems or mechanisms that are dependent on digital signatures or similar integrity protection for email message headers (see also the discussion in Section 11.3). Many conventional uses of PGP and S/MIME are not affected since they are used to sign body parts but not message headers. On the other hand, the developing work on domain keys identified mail (DKIM) [RFC5863] will eventually need to consider this work and vice versa: while this specification does not address or solve the issues raised by DKIM and other signed header mechanisms, the issues will have to be coordinated and resolved eventually if the two sets of protocols are to co-exist. In addition, to the degree to which email addresses appear in PKI (Public Key Infrastructure) certificates [RFC5280], standards addressing such certificates will need to be upgraded to address these internationalized addresses. Those upgrades will need to address questions of spoofing by look-alikes of the addresses themselves.

## 15. Acknowledgments

This document is an update to, and derived from, RFC 4952. This document would have been impossible without the work and contributions acknowledged in it. The present document benefited significantly from discussions in the EAI WG and elsewhere after RFC 4952 was published, especially discussions about the experimental versions of other documents in the internationalized email collection, and from RFC errata on RFC 4952 itself.

Special thanks are due to Ernie Dainow for careful reviews and suggested text in this version and to several IESG members for a careful review and specific suggestions.

## 16. References

### 16.1. Normative References

- |         |  |
|---------|--|
| [ASCII] | American National Standards Institute<br>(formerly United States of America<br>Standards Institute), "USA Code for<br>Information Interchange", ANSI X3.4-1968,<br>1968. |
|         | ANSI X3.4-1968 has been replaced by newer<br>versions with slight modifications, but the<br>1968 version remains definitive for the<br>Internet.                         |

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.
- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5335bis-Hdrs] Yang, A., Steele, S., and N. Freed, "Internationalized Email Headers", September 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-rfc5335bis/>>.
- [RFC5336bis-SMTP] Yao, J. and W. Mao, "SMTP Extension for Internationalized Email Address", August 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-rfc5336bis/>>.
- [RFC5337bis-DSN] Hansen, T., Newman, C., and A. Melnikov, "Internationalized Delivery Status and Disposition Notifications", October 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-rfc5337bis-dsn/>>.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.
- [RFC6152] Klensin, J., Freed, N., Rose, M., and D. Crocker, "SMTP Service Extension for 8-bit MIME Transport", STD 71, RFC 6152, March 2011.

## 16.2. Informative References

- [POPIMAP-downgrade] Fujiwara, K., "Post-delivery Message Downgrading for Internationalized Email Messages", Work in Progress, July 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-popimap-downgrade/>>.

- [RFC0821] Postel, J., "Simple Mail Transfer Protocol", STD 10, RFC 821, August 1982.
- [RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989.
- [RFC1939] Myers, J. and M. Rose, "Post Office Protocol - Version 3", STD 53, RFC 1939, May 1996.
- [RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
- [RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046, November 1996.
- [RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, November 1996.
- [RFC2231] Freed, N. and K. Moore, "MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations", RFC 2231, November 1997.
- [RFC2821] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821, April 2001.
- [RFC3156] Elkins, M., Del Torto, D., Levien, R., and T. Roessler, "MIME Security with OpenPGP", RFC 3156, August 2001.
- [RFC3461] Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, January 2003.
- [RFC3464] Moore, K. and G. Vaudreuil, "An Extensible Message Format for Delivery Status Notifications", RFC 3464, January 2003.
- [RFC3492] Costello, A., "Punycode: A Bootstring encoding of Unicode for Internationalized

- Domain Names in Applications (IDNA)", RFC 3492, March 2003.
- [RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003.
- [RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", RFC 3987, January 2005.
- [RFC4155] Hall, E., "The application/mbox Media Type", RFC 4155, September 2005.
- [RFC4409] Gellens, R. and J. Klensin, "Message Submission for Mail", RFC 4409, April 2006.
- [RFC4690] Klensin, J., Faltstrom, P., Karp, C., and IAB, "Review and Recommendations for Internationalized Domain Names (IDNs)", RFC 4690, September 2006.
- [RFC4952] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", RFC 4952, July 2007.
- [RFC5198] Klensin, J. and M. Padlipsky, "Unicode Format for Network Interchange", RFC 5198, March 2008.
- [RFC5228] Guenther, P. and T. Showalter, "Sieve: An Email Filtering Language", RFC 5228, January 2008.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 5280, May 2008.
- [RFC5335] Abel, Y., "Internationalized Email Headers", RFC 5335, September 2008.
- [RFC5336] Yao, J. and W. Mao, "SMTP Extension for Internationalized Email Addresses", RFC 5336, September 2008.
- [RFC5337] Newman, C. and A. Melnikov,

"Internationalized Delivery Status and Disposition Notifications", RFC 5337, September 2008.

- [RFC5504] Fujiwara, K. and Y. Yoneya, "Downgrading Mechanism for Email Address Internationalization", RFC 5504, March 2009.
- [RFC5721] Gellens, R. and C. Newman, "POP3 Support for UTF-8", RFC 5721, February 2010.
- [RFC5721bis-POP3] Gellens, R., Yao, J., and K. Fujiwara, "POP3 Support for UTF-8", Work in Progress, July 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-rfc5721bis>>.
- [RFC5738] Resnick, P. and C. Newman, "IMAP Support for UTF-8", RFC 5738, March 2010.
- [RFC5738bis-IMAP] Resnick, P., Newman, C., and S. Shen, "IMAP Support for UTF-8", Work in Progress, July 2011, <<https://datatracker.ietf.org/doc/draft-ietf-eai-5738bis/>>.
- [RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 2010.
- [RFC5825] Fujiwara, K. and B. Leiba, "Displaying Downgraded Messages for Email Address Internationalization", RFC 5825, April 2010.
- [RFC5863] Hansen, T., Siegel, E., Hallam-Baker, P., and D. Crocker, "DomainKeys Identified Mail (DKIM) Development, Deployment, and Operations", RFC 5863, May 2010.
- [RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.
- [RFC5892] Faltstrom, P., "The Unicode Code Points and Internationalized Domain Names for Applications (IDNA)", RFC 5892, August 2010.

- [RFC5893] Alvestrand, H. and C. Karp, "Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA)", RFC 5893, August 2010.
- [RFC5894] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale", RFC 5894, August 2010.
- [RFC5983] Gellens, R., "Mailing Lists and Internationalized Email Addresses", RFC 5983, October 2010.
- [RFC5983bis-MailingList] "Mailing Lists and Internationalized Email Addresses", Unwritten waiting for I-D, 2011.
- [RFC6055] Thaler, D., Klensin, J., and S. Cheshire, "IAB Thoughts on Encodings for Internationalized Domain Names", RFC 6055, February 2011.
- [RFC6068] Duerst, M., Masinter, L., and J. Zawinski, "The 'mailto' URI Scheme", RFC 6068, October 2010.
- [Unicode] The Unicode Consortium. The Unicode Standard, Version 5.2.0, defined by:, "The Unicode Standard, Version 5.2.0", (Mountain View, CA: The Unicode Consortium, 2009. ISBN 978-1-936213-00-9)., <<http://www.unicode.org/versions/Unicode5.2.0/>>.
- [Unicode-UAX15] The Unicode Consortium, "Unicode Standard Annex #15: Unicode Normalization Forms", March 2008, <<http://www.unicode.org/reports/tr15/>>.

#### Appendix A. Change Log

[[RFC Editor: Please remove this section prior to publication.]]

##### A.1. Changes between -00 and -01

- o Because there has been no feedback on the mailing list, updated the various questions to refer to this version as well.



- o Reflected RFC Editor erratum #1507 by correcting terminology for headers and header fields and distinguishing between "message headers" and different sorts of headers (e.g., the MIME ones).

#### A.2. Changes between -01 and -02

Note that section numbers in the list that follows may refer to -01 and not -02.

- o Discussion of RFC 5825 ("downgraded display") has been removed per the earlier note and on-list discussion. Any needed discussion about reconstructed messages will need to appear in the IMAP and POP documents. However, the introductory material has been reworded to permit keeping 5504 and 5825 on the list there, without which the back chain would not be complete. For consistency with this change, 5504 and 5825 have been added to the "Obsoletes" list (as far as I know, an Informational spec can obsolete or update Experimental ones, so no downref problem here --JcK).
- o Reference to alternate addresses dropped from (former) Section 7.1.
- o Reference to RFC 5504 added to (former) Section 8 for completeness.
- o Ernie's draft comments added (with some minor edits) to replace the placeholder in (former) Section 9 ("Downgrading in Transit"). It is intended to capture at least an introduction the earlier discussions of algorithmic downgrading generally and ACE/Punycode transformations in particular. Anyone who is unhappy with it should say so and propose alternate text. RSN.
- o In the interest of clarity and consistency with the terminology in Section 4.1, all uses of "final delivery SMTP server" and "final delivery server" have been changed to "final delivery MTA".
- o Placeholder at the end of Section 2 has been removed and the text revised to promise less. The "Document Plan" (Section 5) has been revised accordingly. We need to discuss this at IETF 78 if not sooner.
- o Sections 5 and 6 have been collapsed into one -- there wasn't enough left in the former Section 5 to justify a separate section.
- o Former Section 11.1 has been dropped and the DSN document moved up into the "Document Plan" as suggested earlier.

- o Section 12, "Experimental Targets", has been removed.
- o Updated references for the new version EAI documents and added placeholders for all of the known remaining drafts that will become part of the core EAI series but that have not been written.
- o Inserted an additional clarification about the relationship of these extensions to non-ASCII messages.
- o Changed some normative/informative reference classifications based on review of the new text.
- o Removed references to the pre-EAI documents that were cited for historical context in 4952.
- o Got rid of a remaining pointer to address downgrading in the discussion of an updated MAILTO URI.
- o Minor additional editorial cleanups and tuning.

#### A.3. Changes between -02 and -03

- o Inserted paragraph clarifying the status of the UTF8SMTPbis keyword as a result of discussion prior to and during IETF 79.
- o Adjusted some references including adding an explicit citation of RFC 821.
- o Removed the discussion of the experimental work from an inline aside to a separate section, Section 6.
- o Rewrote the discussion of configuration errors in MX setups to make it clear that they are an issue with forward-pointing addresses only and improved the discussion of backward-pointing addresses.
- o Removed some now-obsolete placeholder notes and resolved the remaining one to a dangling reference.

#### A.4. Changes between -03 and -04

- o Several minor editorial changes.
- o Added a discussion of the relationship to the base email, MIME, and IDNA specifications.

## A.5. Changes between -04 and -05

- o Several more minor editorial changes.

## A.6. Changes between -05 and -06

- o Corrections to more precisely reflect RFC 2119 language requirements and closely-related issues..

## A.7. Changes between -06 and -07

- o Added a new section (now Section 12) to explicitly discuss the changes from the previous version.
- o Removed the discussion of LMTP from Section 11; it is more appropriately placed in the SMTP Extension document (5336bis).

## A.8. Changes between -07 and -08 (after IETF Last Call)

- o Modified Section 7.2 to make the last paragraph less tentative and more clear.
- o Modified Section 8.1 to add an introductory paragraph that clarifies what the IETF does and does not specify about email protocols.

## A.9. Changes between -08 and -09

This version incorporates responses to a last set of public comments and changes made in response to IESG discussion and comments as part of the balloting process.

- o Many small editorial changes made at IESG request.
- o Several other small editorial corrections, removal of uncited reference to LMTP, added a few citations for clarity.

## A.10. Changes between -09 and -10

This version contains additional small editorial changes resulting from IESG comments and review of -09 changes. Some more significant clarifications appear in Section 10.1

## A.11. Changes between -10 and -11

While -10 was approved for publication by the IESG (after IETF Last Call) in September 2010, the document then went into a reference hold in the RFC Editor queue. Issued identified during and after Last

Call for the other three core EAI documents (5335bis, 5336bis, and 5337bis) required reopening this document and making some minor additional changes.

- o Reworded the descriptions of the POP, IMAP, and mailing list documents and moved them to Informative. Notes in the XML of earlier versions of this draft indicate that they were listed as Normative merely as a temporary convenience. Examination and reclassification of them apparently slipped through the cracks.
- o Reclassified the document to standards track to eliminate normative reference problems from other EAI documents.
- o References, other than the two Unicode ones, have been updated for the convenience of reviewers and the RFC Editor. A note has been inserted into the XML requesting that the RFC Editor update the Unicode references to be current at the time of publication.
- o Explicitly notes status of documents obsoleted by this one and moves them to Historic.
- o Updated author contact information.

#### Authors' Addresses

John C KLENSIN  
1770 Massachusetts Ave, #322  
Cambridge, MA 02140  
USA

Phone: +1 617 491 5735  
EMail: john-ietf@jck.com

YangWoo KO  
112-202 Malgeunachim APT. Nae-dong  
Seo-gu, Daejeon 302-981  
Republic of Korea

EMail: yangwooko@gmail.com



Email Address Internationalization  
(EAI)  
Internet-Draft  
Updates: 5322 (if approved)  
Intended status: Standards Track  
Expires: May 3, 2012

K. Fujiwara  
JPRS  
Oct 31, 2011

Post-delivery Message Downgrading for Internationalized Email Messages  
draft-ietf-eai-popimap-downgrade-03.txt

## Abstract

The Email Address Internationalization (UTF8SMTP) extension allows UTF-8 characters in mail header fields. POP and IMAP servers support internationalized email messages. If a POP/IMAP client does not support Email Address Internationalization, POP/IMAP servers cannot send Internationalized Email Headers to the client and cannot remove the message. To avoid the situation, this document describes a conversion mechanism for internationalized Email messages to be traditional message format.

## 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 3, 2012.

## Copyright Notice

Copyright (c) 2011 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

1. Introduction . . . . .	4
2. Terminology . . . . .	5
3. Updating RFC 5322 . . . . .	5
4. New Header Fields Definition . . . . .	6
4.1. Preservation Header Fields . . . . .	6
5. Email Header Fields Downgrading . . . . .	7
5.1. Downgrading Method for Each ABNF Element . . . . .	7
5.1.1. RECEIVED Downgrading . . . . .	7
5.1.2. UNSTRUCTURED Downgrading . . . . .	7
5.1.3. WORD Downgrading . . . . .	7
5.1.4. COMMENT Downgrading . . . . .	7
5.1.5. MIME-VALUE Downgrading . . . . .	7
5.1.6. DISPLAY-NAME Downgrading . . . . .	8
5.1.7. GROUP Downgrading . . . . .	8
5.1.8. MAILBOX Downgrading . . . . .	8
5.1.9. ENCAPSULATION Downgrading . . . . .	9
5.1.10. TYPED-ADDRESS Downgrading . . . . .	9
5.2. Downgrading Method for Each Header Field . . . . .	9
5.2.1. Address Header Fields That Contain <address>s . . . . .	9
5.2.2. Address Header Fields with Typed Addresses . . . . .	10
5.2.3. Downgrading Non-ASCII in Comments . . . . .	10
5.2.4. Message-ID Header Fields . . . . .	10
5.2.5. Received Header Field . . . . .	11
5.2.6. MIME Content Header Fields . . . . .	11
5.2.7. Non-ASCII in <unstructured> . . . . .	11
5.2.8. Non-ASCII in <phrase> . . . . .	11
5.2.9. Other Header Fields . . . . .	11
6. MIME Body-Part Header Field Downgrading . . . . .	12
7. Security Considerations . . . . .	12
8. Implementation Notes . . . . .	13
8.1. RFC 2047 Encoding . . . . .	13
9. IANA Considerations . . . . .	13
9.1. Statement about Downgraded- registration . . . . .	14
9.2. Existing Downgraded- registrations . . . . .	14
9.3. Additional header fields . . . . .	14
10. Acknowledgements . . . . .	15
11. Change History . . . . .	15
11.1. Version 00 . . . . .	15
11.2. Version 01 . . . . .	15
11.3. Version 02 . . . . .	16

11.4. Version 03 . . . . .	16
12. References . . . . .	16
12.1. Normative References . . . . .	16
12.2. Informative References . . . . .	17
Appendix A. Examples . . . . .	17
A.1. Downgrading Example . . . . .	17



## 1. Introduction

Traditional mail systems, which are defined by [RFC5322], allow ASCII characters in mail header field values. The UTF8SMTP extension ([I-D.ietf-eai-frmrwk-4952bis] and [I-D.ietf-eai-rfc5335bis] allows UTF-8 characters in mail header field values.

If a header field contains non-ASCII characters, POP/IMAP servers cannot send Internationalized Email Headers to the client and cannot remove the message. This message downgrading mechanism converts mail header fields to an all-ASCII representation. The POP/IMAP servers can use the downgrading mechanism and send the Internationalized Email message as a traditional form.

[I-D.ietf-eai-rfc5335bis] allows UTF-8 characters to be used in mail header fields and MIME header fields. The message downgrading mechanism specified here describes the conversion method from the internationalized email messages that are defined in [I-D.ietf-eai-frmrwk-4952bis], and [I-D.ietf-eai-rfc5335bis] to the traditional email messages defined in [RFC5322].

There is no good way to convert "From:" and "Sender:" header fields, the draft need to update [RFC5322] to allow empty "From:" and "Sender:" header fields and it is described in Section 3.

Message Downgrading may be implemented in POP server and IMAP server only.

This document tries to define the message downgrading process clearly.

Downgrading consists of the following four parts:

- o Updating RFC 5322
- o New header field definitions
- o Email header field downgrading
- o MIME header field downgrading

In Section 4 of this document, header fields starting with "Downgraded-" are introduced. They preserve the original header fields.

Email header field downgrading is described in Section 5. It generates ASCII-only header fields.

MIME header fields are expanded in [I-D.ietf-eai-rfc5335bis]. MIME header field downgrading is described in Section 6. It generates ASCII-only MIME header fields.

Displaying downgraded messages that originally contained internationalized header fields is out of scope of this document. A POP/IMAP client which does not support UTF8 extension does not know internationalized message format described in [I-D.ietf-eai-rfc5335bis].

## 2. Terminology

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 [RFC2119].

All specialized terms used in this specification are defined in the Email Address Internationalization (EAI) overview [I-D.ietf-eai-frmwk-4952bis], in the mail message specifications [RFC5322], or in the MIME documents [RFC2045] [RFC2047] [RFC2183] [RFC2231]. The terms "ASCII address", "internationalized email address", "non-ASCII address", "il8mail address", "UTF8SMTP", "message", and "mailing list" are used with the definitions from [I-D.ietf-eai-frmwk-4952bis].

This document depends on [I-D.ietf-eai-rfc5335bis]. Key words used in those documents are used in this document, too.

The term "non-ASCII" refers to a UTF-8 string that contains at least one non-ASCII character.

A "UTF8SMTP message" is an email message expanded by [I-D.ietf-eai-rfc5335bis].

## 3. Updating RFC 5322

"From:" header field or "Sender:" header field may contain non-ASCII addresses in internationalized Email messages. These non-ASCII addresses are not allowed in [RFC5322]. The draft proposes that the pop/imap downgrading uses <group> syntax and encodes non-ASCII addresses into <display-name> with empty <group-list> described in Section 5.

This specification redefines "From:", "Sender:", "Resent-From:" and "Resent-Sender:" header fields defined in Section 3.6.2 and 3.6.6 of [RFC5322] to allow <group> in the header fields.

```
from           = "From:" address-list CRLF
resent-from    = "Resent-From:" address-list CRLF
sender         = "Sender:" address CRLF
resent-sender  = "Resent-Sender:" address CRLF
```

#### 4. New Header Fields Definition

New header fields starting with "Downgraded-" are defined here to preserve those mail header field values that contain UTF-8 characters. During downgrading, one new "Downgraded-" header field is added for each mail header field that cannot be passed as-is to a POP/IMAP client that does not support UTF8 extension. The original mail header field is removed. Only those mail header fields that contain non-ASCII characters are affected. The result of this process is a message that is compliant with existing email specifications [RFC5322]. The original internationalized information can be retrieved by examining the "Downgraded-" header fields that were added.

##### 4.1. Preservation Header Fields

New preservation header fields are defined to preserve information that appeared in non-ASCII text in header fields of the incoming message. The values of the new fields holds the original header field value in encoded form. The revised header field syntax is specified as follows:

```
fields          =/ known-downgraded-headers ":"
                  unstructured CRLF

known-downgraded-headers = "Downgraded-" original-headers

original-headers    = "Message-Id" / "Resent-Message-Id" /
                      "In-Reply-To:" / "References:" /
                      "Original-Recipient" / "Final-Recipient"
```

To preserve a header field in a "Downgraded-" header field:

1. Generate a new "Downgraded-" header field whose value is the original header field value.
2. Treat the generated header field content as if it were unstructured, and then apply [RFC2047] encoding with charset UTF-8 as necessary so that the result is ASCII.

3. Remove the original header field.

## 5. Email Header Fields Downgrading

This section defines the conversion method to ASCII for each header field that may contain non-ASCII characters.

[I-D.ietf-eai-rfc5335bis] expands "Received:" header fields; [RFC5322] describes ABNF elements <mailbox>, <word>, <comment>, <unstructured>; [RFC2045] describes ABNF element <value>.

### 5.1. Downgrading Method for Each ABNF Element

Header field downgrading is defined below for each ABNF element. Converting the header field terminates when no non-ASCII characters remain in the header field.

#### 5.1.1. RECEIVED Downgrading

If the header field name is "Received:" and the FOR clause contains a non-ASCII address, remove the FOR clause from the header field. Other parts (not counting <comment>s) should not contain non-ASCII values.

#### 5.1.2. UNSTRUCTURED Downgrading

If the header field has an <unstructured> field that contains non-ASCII characters, apply [RFC2047] encoding with charset UTF-8.

#### 5.1.3. WORD Downgrading

If the header field has any <word> fields that contain non-ASCII characters, apply [RFC2047] encoding with charset UTF-8.

#### 5.1.4. COMMENT Downgrading

If the header field has any <comment> fields that contain non-ASCII characters, apply [RFC2047] encoding with charset UTF-8.

#### 5.1.5. MIME-VALUE Downgrading

If the header field has any <value> elements defined by [RFC2045] and the elements contain non-ASCII characters, encode the <value> elements according to [RFC2231] with charset UTF-8 and leave the language information empty. If the <value> element is <quoted-string> and it contains <CFWS> outside the DQUOTE, remove the <CFWS> before this conversion.

#### 5.1.6. DISPLAY-NAME Downgrading

If the header field has any <address> (<mailbox> or <group>) elements and they have <display-name> elements that contain non-ASCII characters, encode the <display-name> elements according to [RFC2047] with charset UTF-8. DISPLAY-NAME downgrading is the same algorithm as WORD downgrading.

#### 5.1.7. GROUP Downgrading

<group> is defined in Section 3.4 of [RFC5322]. The <group> elements may contain <mailbox>s which contain non-ASCII addresses.

If the header field has any <group> elements that contain <mailbox> elements, and those <mailbox> elements in turn contain non-ASCII addresses, rewrite each <group> element as

```
"Internationalized address removed" display-name ENCODED_WORD ";;"
```

where the <ENCODED\_WORD> is the original <group-list> encoded according to [RFC2047].

#### 5.1.8. MAILBOX Downgrading

The <mailbox> elements have no equivalent format for non-ASCII addresses. If the header field has any <mailbox> elements that contain non-ASCII characters in their <addr-spec> element, rewrite each <addr-spec> element to ASCII-only format. The <addr-spec> element that contains non-ASCII characters may appear in two forms as:

```
"<" addr-spec ">"  
addr-spec
```

Rewrite both as:

```
"Internationalized address " ENCODED-WORD " removed;;"
```

where the <ENCODED-WORD> is the original <addr-spec> encoded according to [RFC2047].

#### 5.1.9. ENCAPSULATION Downgrading

Encapsulate the header field in a "Downgraded-" header field as described in Section 4 as a last resort.

Applying this procedure to "Received:" header field is prohibited. ENCAPSULATION Downgrading is allowed for "Message-ID", "In-Reply-To:", "References:", "Original-Recipient" and "Final-Recipient" header fields.

#### 5.1.10. TYPED-ADDRESS Downgrading

If the header field contains <utf-8-type-addr> and the <utf-8-type-addr> contains raw non-ASCII characters, it is in utf-8-address form. Convert it to utf-8-addr-xtext form. Those forms are described in [I-D.ietf-eai-rfc5337bis-dsn]. COMMENT downgrading is also performed in this case. If the address type is unrecognized and the header field contains non-ASCII characters, then fall back to using ENCAPSULATION downgrading on the entire header field.

### 5.2. Downgrading Method for Each Header Field

Header fields are listed in [RFC4021]. This section describes the downgrading method for each header field.

If the whole mail header field does not contain non-ASCII characters, email header field downgrading is not required. Each header field's downgrading method is described below.

#### 5.2.1. Address Header Fields That Contain <address>s

From:  
Sender:  
To:  
Cc:  
Bcc:  
Reply-To:  
Resent-From:  
Resent-Sender:  
Resent-To:  
Resent-Cc:  
Resent-Bcc:  
Resent-Reply-To:

Return-Path:  
Disposition-Notification-To:

If the header field contains <group> elements that contain non-ASCII addresses, perform COMMENT downgrading, DISPLAY-NAME downgrading, and GROUP downgrading.

If the header field contains <mailbox> elements that contain non-ASCII addresses, perform COMMENT downgrading, DISPLAY-NAME downgrading, and MAILBOX downgrading.

#### 5.2.2. Address Header Fields with Typed Addresses

Original-Recipient:  
Final-Recipient:

If the header field contains non-ASCII characters, perform TYPED-ADDRESS downgrading.

#### 5.2.3. Downgrading Non-ASCII in Comments

Date:  
Resent-Date:  
MIME-Version:  
Content-ID:  
Content-Transfer-Encoding:  
Content-Language:  
Accept-Language:  
Auto-Submitted:

These header fields do not contain non-ASCII characters except in comments. If the header field contains UTF-8 characters in comments, perform COMMENT downgrading.

#### 5.2.4. Message-ID Header Fields

Message-ID:  
Resent-Message-ID:  
In-Reply-To:  
References:

Perform ENCAPSULATION Downgrading.

#### 5.2.5. Received Header Field

Received:

Perform COMMENT downgrading and RECEIVED downgrading.

#### 5.2.6. MIME Content Header Fields

Content-Type:

Content-Disposition:

Perform MIME-VALUE downgrading and COMMENT downgrading.

#### 5.2.7. Non-ASCII in <unstructured>

Subject:

Comments:

Content-Description:

Perform UNSTRUCTURED downgrading.

#### 5.2.8. Non-ASCII in <phrase>

Keywords:

Perform WORD downgrading.

#### 5.2.9. Other Header Fields

There are other header fields that contain non-ASCII characters. They are user-defined and missing from this document, or future defined header fields. They are treated as "Optional Fields" and their field value are treated as unstructured described in Section 3.6.8 of [RFC5322].

Perform UNSTRUCTURED downgrading.

If the software understands the header field's structure and a downgrading algorithm other than UNSTRUCTURED is applicable, that software SHOULD use that algorithm; UNSTRUCTURED downgrading is used as a last resort.

Mailing list header fields (those that start in "List-") are part of this category.



## 6. MIME Body-Part Header Field Downgrading

MIME body-part header fields may contain non-ASCII characters [I-D.ietf-eai-rfc5335bis]. This section defines the conversion method to ASCII-only header fields for each MIME header field that contains non-ASCII characters. Parse the message body's MIME structure at all levels and check each MIME header field to see whether it contains non-ASCII characters. If the header field contains non-ASCII characters in the header field value, the header field is a target of the MIME body-part header field's downgrading. Each MIME header field's downgrading method is described below. COMMENT downgrading, MIME-VALUE downgrading, and UNSTRUCTURED downgrading are described in Section 5.

### Content-ID:

The "Content-ID:" header field does not contain non-ASCII characters except in comments. If the header field contains UTF-8 characters in comments, perform COMMENT downgrading.

### Content-Type:

### Content-Disposition:

Perform MIME-VALUE downgrading and COMMENT downgrading.

Content-Description: Perform UNSTRUCTURED downgrading.

## 7. Security Considerations

Existing clients do not know new From: and Sender: header fields syntax updated by Section 3 and may get wrong when they confront <group> syntax in From: and Sender: fields.

A downgraded message's header fields contain ASCII characters only. But they still contain MIME-encapsulated header fields that contain non-ASCII UTF-8 characters. Furthermore, the body part may contain UTF-8 characters. Implementations parsing Internet messages need to accept UTF-8 body parts and UTF-8 header fields that are MIME-encoded. Thus, this document inherits the security considerations of MIME-encoded header fields ([RFC2047] and [RFC3629]).

Rewriting header fields increases the opportunities for undetected spoofing by malicious senders. However, rewritten header fields are preserved into Downgraded-\* header fields, and parsing Downgraded-\* header fields enables the detection of spoofing caused by downgrading.

The techniques described here invalidate methods that depend on digital signatures over any part of the message, which includes the

top-level header fields and body-part header fields. Depending on the specific message being downgraded, the following techniques are likely to break: DomainKeys Identified Mail (DKIM), and possibly S/MIME and Pretty Good Privacy (PGP). The two obvious mitigations are to stick to 7-bit transport when using these techniques (as most/all of them presently require) or to make sure to have UTF8SMTP end-to-end when needed.

While information in any email header field should usually be treated with some suspicion, current email systems commonly employ various mechanisms and protocols to make the information more trustworthy. Currently, information in the new Downgraded-\* header fields is usually not inspected by these mechanisms, and may be even less trustworthy than the traditional header fields. Note that the Downgraded-\* header fields could have been inserted with malicious intent (and with content unrelated to the traditional header fields).

See the "Security Considerations" section in [I-D.ietf-eai-frmwrk-4952bis] for more discussion.

## 8. Implementation Notes

### 8.1. RFC 2047 Encoding

While [RFC2047] has a specific algorithm to deal with whitespace in adjacent encoded words, there are a number of deployed implementations that fail to implement the algorithm correctly. As a result, whitespace behavior is somewhat unpredictable in practice when multiple encoded words are used. While RFC 5322 states that implementations SHOULD limit lines to not more than 78 characters, implementations MAY choose to allow overly long encoded words in order to work around faulty [RFC2047] implementations. Implementations that choose to do so SHOULD have an optional mechanism to limit line length to 78 characters.

## 9. IANA Considerations

[[RFC Editor: Please change "should now be" and "should be" to "have been" when the IANA actions are complete.]]

[[ Notes in draft: this section is not finished, to be reviewed with IANA. ]]

Following instructions in the now-obsolete [RFC5504], IANA has made a series of entries in the the Permanent Message Header Field registry. Those registrations should now be changed as follows:

### 9.1. Statement about Downgraded- registration

The statement about refusing any "Downgraded-" registrations should be updated to refer to this document and to provide for registering such fields as specified in Section 9.3.

[[ Note in draft: The restriction may become useless if unknown header fields may be treated as unstructured. ]]

### 9.2. Existing Downgraded- registrations

Individual existing registrations for

- Downgraded-Bcc
- Downgraded-Cc
- Downgraded-Disposition-Notification-To
- Downgraded-From
- Downgraded-Mail-From
- Downgraded-Rcpt-To
- Downgraded-Reply-To
- Downgraded-Resent-Bcc
- Downgraded-Resent-Cc
- Downgraded-Resent-From
- Downgraded-Resent-Reply-To
- Downgraded-Resent-Sender
- Downgraded-Resent-To
- Downgraded-Return-Path
- Downgraded-Sender
- Downgraded-To

should be updated to replace "experimental" with "obsoleted" and to reference this document.

### 9.3. Additional header fields

The following header fields should be registered in the Permanent Message Header Field registry, in accordance with the procedures set out in [RFC3864].

Header field name: Downgraded-Message-Id  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 4)

Header field name: Downgraded-In-Reply-To  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 4)

Header field name: Downgraded-References  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 4)

Header field name: Downgraded-Original-Recipient  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 4)

Header field name: Downgraded-Final-Recipient  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 4)

## 10. Acknowledgements

This document draws heavily from the experimental in-transit message downgrading procedure described in RFC 5504 [RFC5504]. The contribution of the co-author of that earlier document, Y. Yoneya, are gratefully acknowledged.

## 11. Change History

This section is used for tracking the update of this document. Will be removed after finalize.

### 11.1. Version 00

- o Initial version
- o Imported header field downgrading from RFC 5504

### 11.2. Version 01

- o same as Version 00

### 11.3. Version 02

- o Added updating RFC 5322 to allow <group> syntax in From: and Sender
- o Added GROUP Downgrading

### 11.4. Version 03

- o Replaced <utf8-addr-spec> with <addr-spec>
- o Added updating RFC 5322 to allow <group> syntax in From: and Sender
- o Added one sentence in Security considerations
- o Updated IANA considerations

## 12. References

### 12.1. Normative References

- [I-D.ietf-eai-frmwrk-4952bis] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", draft-ietf-eai-frmwrk-4952bis-12 (work in progress), October 2011.
- [I-D.ietf-eai-rfc5335bis] Yang, A., Steele, S., and N. Freed, "Internationalized Email Headers", draft-ietf-eai-rfc5335bis-13 (work in progress), October 2011.
- [I-D.ietf-eai-rfc5337bis-dsn] Hansen, T., Newman, C., and A. Melnikov, "Internationalized Delivery Status and Disposition Notifications", draft-ietf-eai-rfc5337bis-dsn-05 (work in progress), October 2011.
- [RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
- [RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-

ASCII Text", RFC 2047, November 1996.

[RFC2183] Troost, R., Dorner, S., and K. Moore, "Communicating Presentation Information in Internet Messages: The Content-Disposition Header Field", RFC 2183, August 1997.

[RFC2231] Freed, N. and K. Moore, "MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages, and Continuations", RFC 2231, November 1997.

[RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[RFC4021] Klyne, G. and J. Palme, "Registration of Mail and MIME Header Fields", RFC 4021, March 2005.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", BCP 90, RFC 3864, September 2004.

## 12.2. Informative References

[RFC5504] Fujiwara, K. and Y. Yoneya, "Downgrading Mechanism for Email Address Internationalization", RFC 5504, March 2009.

## Appendix A. Examples

### A.1. Downgrading Example

This appendix shows an message downgrading example. Consider a received mail message where:

- o The sender address is a non-ASCII address, "NON-ASCII-local@example.com". Its display-name is "DISPLAY-local".
- o The "To:" header field contains two non-ASCII addresses, "NON-ASCII-remote1@example.net" and "NON-ASCII-remote2@example.com" Its display-names are "DISPLAY-remote1" and "DISPLAY-remote2".
- o The "Cc:" header field contains a non-ASCII address, "NON-ASCII-remote3@example.org". Its display-name is "DISPLAY-remote3".
- o Four display names contain non-ASCII characters.
- o The Subject header field is "NON-ASCII-SUBJECT", which contains non-ASCII characters.
- o The "Message-Id:" header field contains "NON-ASCII-MESSAGE\_ID", which contains non-ASCII characters.
- o There is an unknown header field "X-Unknown-Header" which contains non-ASCII characters.

```
Return-Path: <NON-ASCII-local@example.com>
Received: from ... by ... for <NON-ASCII-remote1@example.net>
Received: from ... by ... for <NON-ASCII-remote1@example.net>
From: DISPLAY-local <NON-ASCII-local@example.com>
To: DISPLAY-remote1 <NON-ASCII-remote1@example.net>,
    DISPLAY-remote2 <NON-ASCII-remote2@example.com>
Cc: DISPLAY-remote3 <NON-ASCII-remote3@example.org>
Subject: NON-ASCII-SUBJECT
Date: DATE
Message-Id: NON-ASCII-MESSAGE_ID
Mime-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
X-Unknown-Header: NON-ASCII-CHARACTERS
```

MAIL\_BODY

Figure 1: Received message in a mail drop

The downgraded message is shown in Figure 2. "Return-Path:", "From:", "To:" and "Cc:" header fields are rewritten. "Subject:" and

"X-Unknown-Header:" header fields are encoded using [RFC2047].  
"Message-Id:" header field is encapsulated as  
"Downgraded-Message-Id:" header field.

Return-Path: Internationalized address  
=?UTF-8?Q?NON-ASCII-local@example.com?= removed;;  
Received: from ... by ...  
Received: from ... by ...  
From: =?UTF-8?Q?DISPLAY-local?= Internationalized address  
=?UTF-8?Q?NON-ASCII-local@example.com?= removed;;  
To: =?UTF-8?Q?DISPLAY-remote1?= Internationalized address  
=?UTF-8?Q?NON-ASCII-remote1@example.net?= removed;;,  
=?UTF-8?Q?DISPLAY-remote2?= Internationalized address  
=?UTF-8?Q?NON-ASCII-remote2@example.com?= removed;;,  
Cc: =?UTF-8?Q?DISPLAY-remote3?= Internationalized address  
=?UTF-8?Q?NON-ASCII-remote3@example.org?= removed;;  
Subject: =?UTF-8?Q?NON-ASCII-SUBJECT?=  
Date: DATE  
Downgraded-Message-Id: =?UTF-8?Q?MESSAGE\_ID?=  
Mime-Version: 1.0  
Content-Type: text/plain; charset="UTF-8"  
Content-Transfer-Encoding: 8bit  
X-Unknown-Header: =?UTF-8?Q?NON-ASCII-CHARACTERS?=  
  
MAIL\_BODY

Figure 2: Downgraded message

#### Author's Address

Kazunori Fujiwara  
Japan Registry Services Co., Ltd.  
Chiyoda First Bldg. East 13F, 3-8-1 Nishi-Kanda  
Chiyoda-ku, Tokyo 101-0065  
Japan

Phone: +81 3 5215 8451  
EMail: fujiwara@wide.ad.jp, fujiwara@jprs.co.jp





Email Address Internationalization  
(EAI)  
Internet-Draft  
Intended status: Standards Track  
Expires: April 25, 2013

K. Fujiwara  
JPRS  
Oct 22, 2012

Post-delivery Message Downgrading for Internationalized Email Messages  
draft-ietf-eai-popimap-downgrade-08.txt

#### Abstract

The Email Address Internationalization (SMTPUTF8) extension to SMTP allows UTF-8 characters in mail header fields. Upgraded POP and IMAP servers support internationalized Email messages. If a POP/IMAP client does not support Email Address Internationalization, POP/IMAP servers cannot deliver Internationalized Email Headers to the client and cannot remove the message. To avoid the situation, this document describes a conversion mechanism for internationalized Email messages to be in traditional message format. In the process, message elements requiring internationalized treatment are recoded or removed and receivers are able to know that they received messages containing such elements even if they cannot process the internationalized elements.

#### 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 April 25, 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

1.	Introduction . . . . .	4
1.1.	Problem statement . . . . .	4
1.2.	Possible solutions . . . . .	4
1.3.	Approach taken in this specification . . . . .	4
2.	Terminology . . . . .	6
3.	Email Header Fields Downgrading . . . . .	6
3.1.	Downgrading Method for Each ABNF Element . . . . .	6
3.1.1.	UNSTRUCTURED Downgrading . . . . .	6
3.1.2.	WORD Downgrading . . . . .	6
3.1.3.	COMMENT Downgrading . . . . .	6
3.1.4.	MIME-VALUE Downgrading . . . . .	7
3.1.5.	DISPLAY-NAME Downgrading . . . . .	7
3.1.6.	DOMAIN Downgrading . . . . .	7
3.1.7.	GROUP Downgrading . . . . .	7
3.1.8.	MAILBOX Downgrading . . . . .	8
3.1.9.	TYPED-ADDRESS Downgrading . . . . .	8
3.1.10.	ENCAPSULATION: A Last Resort . . . . .	8
3.2.	Downgrading Method for Each Header Field . . . . .	10
3.2.1.	Address Header Fields That Contain <address>s . . . . .	10
3.2.2.	Downgrading Non-ASCII in Comments . . . . .	11
3.2.3.	Message-ID Header Fields . . . . .	11
3.2.4.	Received Header Field . . . . .	11
3.2.5.	MIME Content Header Fields . . . . .	12
3.2.6.	Non-ASCII in <unstructured> . . . . .	12
3.2.7.	Non-ASCII in <phrase> . . . . .	12
3.2.8.	Other Header Fields . . . . .	12
4.	MIME Downgrading . . . . .	12
4.1.	MIME Body-Part Header Field Downgrading . . . . .	13
4.2.	Delivery Status Notification downgrading . . . . .	13
5.	Security Considerations . . . . .	13
6.	Implementation Notes . . . . .	14
6.1.	RFC 2047 Encoding . . . . .	14
7.	IANA Considerations . . . . .	15
7.1.	Obsolescence of Existing Downgraded-* Header Fields . . . . .	15
7.2.	Registration of New Downgraded-* Header Fields . . . . .	15
8.	Acknowledgements . . . . .	16
9.	References . . . . .	16

9.1. Normative References . . . . .	16
9.2. Informative References . . . . .	18
Appendix A. Examples . . . . .	18
A.1. Downgrading Example . . . . .	18
Appendix B. Change History . . . . .	20
B.1. Version 00 . . . . .	20
B.2. Version 01 . . . . .	20
B.3. Version 02 . . . . .	20
B.4. Version 03 . . . . .	21
B.5. Version 04 . . . . .	21
B.6. Version 05 . . . . .	21
B.7. Version 06 . . . . .	21
B.8. Version 07 . . . . .	22
B.9. Version 08 . . . . .	22

## 1. Introduction

### 1.1. Problem statement

Traditional (legacy) mail systems, which are defined by [RFC5322] and other specifications, allow only ASCII characters in mail header field values. The SMTPUTF8 extension ([RFC6530], [RFC6531] and [RFC6532]) allow raw UTF-8 in those mail header fields.

If a header field contains non-ASCII strings, POP/IMAP servers cannot deliver Internationalized Email Headers to legacy clients which does not send UTF8 command or UTF8 capability, and because they have no obvious or standardized way to explain what is going on to those clients, cannot even safely discard the message.

### 1.2. Possible solutions

There are four plausible approaches to the problem, with the preferred one depending on the particular circumstances and relationship among the delivery SMTP server, the mail store, the POP or IMAP server, and the users and their MUA clients:

1. If the delivery MTA has sufficient knowledge about the POP and/or IMAP servers and clients being used, the message may be rejected as undeliverable.
2. The message may be downgraded by the POP or IMAP server, in a way that preserves maximum information at the expense of some complexity, and does not create security or operational problems in the mail system.
3. Some intermediate downgrading may be applied that balances more information loss against lower complexity and greater ease of implementation.
4. The POP or IMAP server may fabricate a message whose intent is to notify the client that an internationalized message is waiting but cannot be delivered until an upgraded client is available.

### 1.3. Approach taken in this specification

This specification describes the second of those options. It is worth noticing that, at least in the general case, none of these options preserve sufficient information to guarantee that it is possible to reply to an incoming message without loss of information, so the choice may be considered to be among "least bad" options. While this document specifies a well designed mechanism, it is only an interim solution while clients are being upgraded

[I-D.ietf-eai-rfc5721bis] [I-D.ietf-eai-5738bis].

This message downgrading mechanism converts mail header fields to an all-ASCII representation. The POP/IMAP servers can use the downgrading mechanism and deliver the Internationalized Email message as a traditional form. Receivers can know they received some internationalized messages or some unknown or broken messages.

[RFC6532] allows UTF-8 characters to be used in mail header fields and MIME header fields. [RFC6531] allows UTF-8 characters to be used in some trace header fields. The message downgrading mechanism specified here describes the conversion method from the internationalized messages that are defined in [RFC6530], and [RFC6532] to the traditional email messages defined in [RFC5322].

This document provides a precise definition of the minimum-information-loss message downgrading process.

Downgrading consists of the following three parts:

- o New header field definitions
- o Email header field downgrading
- o MIME header field downgrading

Email header field downgrading is described in Section 3. It generates ASCII-only header fields.

In Section 3.1.10 of this document, header fields starting with "Downgraded-" are introduced. They preserve the information that appeared in the original header fields.

The definition of MIME header fields in Internationalized Email Messages is described in [RFC6532]. MIME header field downgrading is described in Section 4.1. It generates ASCII-only MIME header fields.

Displaying downgraded messages that originally contained internationalized header fields is out of scope of this document. A POP/IMAP client which does not support UTF8 extensions as defined for POP3 [UTF8 command] and IMAP ["ENABLE UTF8=ACCEPT" command] does not know internationalized message format described in [RFC6532].

## 2. Terminology

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 [RFC2119].

All specialized terms used in this specification are defined in the Overview and Framework for Internationalized Email [RFC6530], in the mail message specifications [RFC5322], or in the MIME documents [RFC2045] [RFC2047] [RFC2183] [RFC2231]. The terms "U-label", "A-label" and "IDNA" are used with the definitions from [RFC5890]. The terms "ASCII address", "non-ASCII address", "SMTPUTF8", "message", "internationalized message" are used with the definitions from [RFC6530]. The term "non-ASCII string" is used with the definitions from [RFC6532].

## 3. Email Header Fields Downgrading

This section defines the conversion method to ASCII for each header field that may contain non-ASCII strings. Section 3.1 describes rewriting methods for each ABNF element. Section 3.2 describes rewriting methods for each header field.

### 3.1. Downgrading Method for Each ABNF Element

Header field downgrading is defined below for each ABNF element. Converting the header field terminates when no non-ASCII strings remain in the header field.

[RFC5322] describes ABNF elements <group>, <mailbox>, <unstructured>, <word>, <comment>, <display-name>. [RFC2045] describes ABNF element <value>. <domain> is updated to allow non-ASCII characters in Section 3.3 of [RFC6531] and Section 3.2 of [RFC6532].

#### 3.1.1. UNSTRUCTURED Downgrading

If the header field has an <unstructured> field that contains non-ASCII strings, apply [RFC2047] encoding with charset UTF-8.

#### 3.1.2. WORD Downgrading

If the header field has any <word> fields that contain non-ASCII strings, apply [RFC2047] encoding with charset UTF-8.

#### 3.1.3. COMMENT Downgrading

If the header field has any <comment> fields that contain non-ASCII strings, apply [RFC2047] encoding with charset UTF-8.

#### 3.1.4. MIME-VALUE Downgrading

If the header field has any <value> elements defined by [RFC2045] and the elements contain non-ASCII strings, encode the <value> elements according to [RFC2231] with charset UTF-8 and leave the language information empty. If the <value> element is <quoted-string> and it contains <CFWS> outside the DQUOTE, remove the <CFWS> before this conversion.

#### 3.1.5. DISPLAY-NAME Downgrading

If the header field has any <address> (<mailbox> or <group>) elements and they have <display-name> elements that contain non-ASCII strings, encode the <display-name> elements according to [RFC2047] with charset UTF-8. DISPLAY-NAME downgrading is the same algorithm as WORD downgrading.

#### 3.1.6. DOMAIN Downgrading

If the header field has any <domain> elements that contain U-labels, rewrite the non-ASCII domain name into ASCII domain name using A-labels as specified in IDNA [RFC5891].

#### 3.1.7. GROUP Downgrading

<group> is defined in Section 3.4 of [RFC5322]. The <group> elements may contain <mailbox>es which contain non-ASCII addresses.

If a <group> element contains <mailbox> elements and one of <mailbox>es contains a non-ASCII <local-part>, rewrite the <group> element as

```
display-name " " ENCODED_WORD " ;"
```

where the <ENCODED\_WORD> is the original <group-list> encoded according to [RFC2047].

Otherwise, the <group> element does not contain non-ASCII <local-part>. If the <group> element contain non-ASCII <mailbox>es, they contains non-ASCII domain names. Rewrite the non-ASCII domain names into ASCII domain names using A-labels as specified in IDNA [RFC5891]. Generated <mailbox>es contain ASCII addresses only.



### 3.1.8. MAILBOX Downgrading

If the <local-part> of the <mailbox> element does not contain non-ASCII characters, the <domain> element contains non-ASCII characters. Rewrite the non-ASCII domain name into ASCII domain name using A-labels as specified in IDNA [RFC5891].

Otherwise, the <local-part> contains non-ASCII characters. The non-ASCII <local-part> has no equivalent format for ASCII addresses. The <addr-spec> element that contains non-ASCII strings may appear in two forms as:

```
"<" addr-spec ">"  
addr-spec
```

Rewrite both as:

```
ENCODED-WORD " :;"
```

where the <ENCODED-WORD> is the original <addr-spec> encoded according to [RFC2047].

### 3.1.9. TYPED-ADDRESS Downgrading

If the header field contains <utf-8-type-addr> and the <utf-8-type-addr> contains raw non-ASCII strings, it is in utf-8-address form. Convert it to utf-8-addr-xtext form. Those forms are described in [RFC6533]. COMMENT downgrading is also performed in this case. If the address type is unrecognized and the header field contains non-ASCII strings, then fall back to using ENCAPSULATION on the entire header field specified in Section 3.1.10.

### 3.1.10. ENCAPSULATION: A Last Resort

As a last resort when header fields cannot be converted as discussed in the previous section, the fields are deleted and replaced by specialized new header fields. Those fields are defined to preserve, in encoded form, as much information as possible from the header field values of the incoming message. The syntax of these new header fields is:

```
fields                                =/ downgraded

downgraded = "Downgraded-Message-Id:"      unstructured CRLF /
             "Downgraded-Resent-Message-Id:" unstructured CRLF /
             "Downgraded-In-Reply-To:"      unstructured CRLF /
             "Downgraded-References:"       unstructured CRLF /
             "Downgraded-Original-Recipient:" unstructured CRLF /
             "Downgraded-Final-Recipient:"  unstructured CRLF
```

Applying this procedure to "Received:" header field is prohibited. ENCAPSULATION Downgrading is allowed for "Message-ID", "In-Reply-To:", "References:", "Original-Recipient" and "Final-Recipient" header fields.

To preserve a header field in a "Downgraded-" header field:

1. Generate a new header field.
  - \* The field name is a concatenation of "Downgraded-" and the original field name.
  - \* The initial new field value is the original header field value.
2. Treat the initial new header field value as if it were unstructured, and then apply [RFC2047] encoding with charset UTF-8 as necessary so that the resulting new header field value is completely in ASCII.
3. Remove the original header field.

### 3.2. Downgrading Method for Each Header Field

[RFC4021] establishes a registry of header fields. This section describes the downgrading method for each header field.

If the whole mail header field does not contain non-ASCII strings, email header field downgrading is not required. Each header field's downgrading method is described below.

#### 3.2.1. Address Header Fields That Contain <address>s

From:  
Sender:  
To:  
Cc:  
Bcc:  
Reply-To:  
Resent-From:  
Resent-Sender:  
Resent-To:  
Resent-Cc:  
Resent-Bcc:  
Resent-Reply-To:  
Return-Path:  
Disposition-Notification-To:

If the header field contains non-ASCII characters, first perform COMMENT downgrading and DISPLAY-NAME downgrading as described in the corresponding subsections of Section 3.1. If the header field still contains non-ASCII characters after that, do the following two steps:

1. If the header field contains <group> elements that contain non-ASCII addresses, perform GROUP downgrading on those elements.
2. If the header field contains <mailbox> elements that contain non-ASCII addresses, perform MAILBOX downgrading on those elements.

This procedure may generate empty <group> elements in "From:", "Sender:" and "Reply-To:" header fields.  
[I-D.leiba-5322upd-from-group] updates [RFC5322] to allow (empty) <group> elements in "From:", "Sender:" and "Reply-To:" header fields.

### 3.2.2. Downgrading Non-ASCII in Comments

Date:  
Resent-Date:  
MIME-Version:  
Content-ID:  
Content-Transfer-Encoding:  
Content-Language:  
Accept-Language:  
Auto-Submitted:

These header fields do not contain non-ASCII strings except in comments. If the header field contains UTF-8 characters in comments, perform COMMENT downgrading.

### 3.2.3. Message-ID Header Fields

Message-ID:  
Resent-Message-ID:  
In-Reply-To:  
References:

Perform ENCAPSULATION as specified in Section 3.1.10.

### 3.2.4. Received Header Field

Received:

If <domain> elements or <mailbox> elements contains U-labels, perform DOMAIN downgrading specified in Section 3.1.6. Comments may contain non-ASCII strings, perform COMMENT downgrading.

After the DOMAIN downgrading and the COMMENT downgrading, if the FOR clause contains a non-ASCII <local-part>, remove the "FOR" clause. If the ID clause contains a non-ASCII values, remove the "ID" clause.

### 3.2.5. MIME Content Header Fields

Content-Type:  
Content-Disposition:

Perform MIME-VALUE downgrading and COMMENT downgrading.

### 3.2.6. Non-ASCII in <unstructured>

Subject:  
Comments:  
Content-Description:

Perform UNSTRUCTURED downgrading.

### 3.2.7. Non-ASCII in <phrase>

Keywords:

Perform WORD downgrading.

### 3.2.8. Other Header Fields

There are other header fields that contain non-ASCII strings. They are user-defined and missing from this document, or future defined header fields. They are treated as "Optional Fields" and their field values are treated as unstructured described in Section 3.6.8 of [RFC5322].

Perform UNSTRUCTURED downgrading.

If the software understands the header field's structure and a downgrading algorithm other than UNSTRUCTURED is applicable, that software SHOULD use that algorithm; UNSTRUCTURED downgrading is used as a last resort.

Mailing list header fields (those that start in "List-") are part of this category.

## 4. MIME Downgrading

Both MIME Body-Part header fields and contents of a delivery status notification may contain non-ASCII characters.

#### 4.1. MIME Body-Part Header Field Downgrading

MIME body-part header fields may contain non-ASCII strings [RFC6532]. This section defines the conversion method to ASCII-only header fields for each MIME header field that contains non-ASCII strings. Parse the message body's MIME structure at all levels and check each MIME header field to see whether it contains non-ASCII strings. If the header field contains non-ASCII strings in the header field value, the header field is a target of the MIME body-part header field's downgrading. Each MIME header field's downgrading method is described below. COMMENT downgrading, MIME-VALUE downgrading, and UNSTRUCTURED downgrading are described in Section 3.

Content-ID:

The "Content-ID:" header field does not contain non-ASCII strings except in comments. If the header field contains UTF-8 characters in comments, perform COMMENT downgrading.

Content-Type:

Content-Disposition:

Perform MIME-VALUE downgrading and COMMENT downgrading.

Content-Description:

Perform UNSTRUCTURED downgrading.

#### 4.2. Delivery Status Notification downgrading

If the message contains a delivery status notification defined at Section 6 of [RFC3461], perform the following tests and conversions.

If there are "Original-Recipient:" and "Final-Recipient:" header fields, and the header fields contain non-ASCII strings, perform TYPED-ADDRESS downgrading.

### 5. Security Considerations

The purpose of post-delivery message downgrading is to allow POP/IMAP servers to deliver internationalized messages to traditional POP/IMAP clients and permit the clients to display those messages. Users who receive such messages can know that they were internationalized. It does not permit receivers to read the messages in their original form and, in general, will not permit generating replies, at least without significant user intervention.

A downgraded message's header fields contain ASCII characters only. But they still contain MIME-encapsulated header fields that contain non-ASCII strings. Furthermore, the body part may contain UTF-8 characters. Implementations parsing Internet messages need to accept

UTF-8 body parts and UTF-8 header fields that are MIME-encoded. Thus, this document inherits the security considerations of MIME-encoded header fields ([RFC2047] and [RFC3629]).

Rewriting header fields increases the opportunities for undetected spoofing by malicious senders. However, the rewritten header field values are preserved in equivalent MIME form or in newly defined header fields for which traditional MUAs have no special processing procedures.

The techniques described here invalidate methods that depend on digital signatures over any part of the message, which includes the top-level header fields and body-part header fields. Depending on the specific message being downgraded, at least the following techniques are likely to break: DomainKeys Identified Mail (DKIM), and possibly S/MIME and Pretty Good Privacy (PGP). The downgrade mechanism SHOULD NOT remove signatures even if the signatures will fail validation after downgrading. As much of the information as possible from the original message SHOULD be preserved.

While information in any email header field should usually be treated with some suspicion, current email systems commonly employ various mechanisms and protocols to make the information more trustworthy. Information in the new Downgraded-\* header fields is not inspected by traditional MUAs, and may be even less trustworthy than the traditional header fields. Note that the Downgraded-\* header fields could have been inserted with malicious intent (and with content unrelated to the traditional header fields), however traditional MUAs do not parse Downgraded-\* header fields.

In addition, if an Authentication-Results header field [RFC5451] is present, traditional MUAs may treat that the digital signatures are valid.

See the "Security Considerations" section in [I-D.leiba-5322upd-from-group] and [RFC6530] for more discussion.

## 6. Implementation Notes

### 6.1. RFC 2047 Encoding

While [RFC2047] has a specific algorithm to deal with whitespace in adjacent encoded words, there are a number of deployed implementations that fail to implement the algorithm correctly. As a result, whitespace behavior is somewhat unpredictable in practice when multiple encoded words are used. While RFC 5322 states that implementations SHOULD limit lines to not more than 78 characters, implementations MAY choose to allow overly long encoded words in

order to work around faulty [RFC2047] implementations. Implementations that choose to do so SHOULD have an optional mechanism to limit line length to 78 characters.

## 7. IANA Considerations

[[RFC Editor: Please change "is asked to" to "has" (and change the verb correspondingly) when the IESG approval and IANA actions are complete.]]

[RFC5504] specified that no new header fields be registered that begin with "Downgraded-". That restriction is now lifted, and this document makes a new set of registrations, replacing the experimental fields with standard ones.

### 7.1. Obsolescence of Existing Downgraded-\* Header Fields

The "Downgraded-\*" header fields that were registered as experimental fields in [RFC5504] are no longer in use. IANA is asked to change the status from "experimental" to "obsoleted" for every name in the Permanent Message Header Field registry that begins with "Downgraded-".

### 7.2. Registration of New Downgraded-\* Header Fields

[[RFC Editor: Please change "should be" to "have been" when the IANA actions are complete.]]

The following header fields should be registered in the Permanent Message Header Field registry, in accordance with the procedures set out in [RFC3864].

Header field name: Downgraded-Message-Id  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 3.1.10)

Header field name: Downgraded-In-Reply-To  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 3.1.10)



Header field name: Downgraded-References  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 3.1.10)

Header field name: Downgraded-Original-Recipient  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 3.1.10)

Header field name: Downgraded-Final-Recipient  
Applicable protocol: mail  
Status: standard  
Author/change controller: IETF  
Specification document(s): This document (Section 3.1.10)

## 8. Acknowledgements

This document draws heavily from the experimental in-transit message downgrading procedure described in RFC 5504 [RFC5504]. The contribution of the co-author of that earlier document, Y. Yoneya, are gratefully acknowledged. Significant comments and suggestions were received from John Klensin, Barry Leiba, Randall Gellens, Pete Resnick, Martin J. Durst, and other WG participants.

## 9. References

### 9.1. Normative References

- [RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.
- [RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, November 1996.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2183] Troost, R., Dorner, S., and K. Moore, "Communicating Presentation

- Information in Internet Messages: The Content-Disposition Header Field", RFC 2183, August 1997.
- [RFC2231] Freed, N. and K. Moore, "MIME Parameter Value and Encoded Word Extensions: Character Sets, Languages , and Continuations", RFC 2231, November 1997.
- [RFC3461] Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, January 2003.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.
- [RFC3864] Klyne, G., Nottingham, M., and J. Mogul, "Registration Procedures for Message Header Fields", BCP 90, RFC 3864, September 2004.
- [RFC4021] Klyne, G. and J. Palme, "Registration of Mail and MIME Header Fields", RFC 4021, March 2005.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5890] Klensin, J., "Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework", RFC 5890, August 2010.
- [RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.
- [RFC6530] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", RFC 6530, February 2012.
- [RFC6531] Yao, J. and W. Mao, "SMTP Extension for Internationalized Email", RFC 6531, February 2012.

- [RFC6532] Yang, A., Steele, S., and N. Freed, "Internationalized Email Headers", RFC 6532, February 2012.
- [RFC6533] Hansen, T., Newman, C., and A. Melnikov, "Internationalized Delivery Status and Disposition Notifications", RFC 6533, February 2012.
- [I-D.leiba-5322upd-from-group] Leiba, B., "Update to Internet Message Format to Allow Group Syntax in the "From:" and "Sender:" Header Fields", draft-leiba-5322upd-from-group-06 (work in progress), October 2012.
- [I-D.ietf-eai-rfc5721bis] Gellens, R., Newman, C., Yao, J., and K. Fujiwara, "POP3 Support for UTF-8", draft-ietf-eai-rfc5721bis-08 (work in progress), October 2012.
- [I-D.ietf-eai-5738bis] Resnick, P., Newman, C., and S. Shen, "IMAP Support for UTF-8", draft-ietf-eai-5738bis-09 (work in progress), August 2012.

## 9.2. Informative References

- [RFC5451] Kucherawy, M., "Message Header Field for Indicating Message Authentication Status", RFC 5451, April 2009.
- [RFC5504] Fujiwara, K. and Y. Yoneya, "Downgrading Mechanism for Email Address Internationalization", RFC 5504, March 2009.

## Appendix A. Examples

### A.1. Downgrading Example

This appendix shows an message downgrading example. Consider a received mail message where:

- o The sender address is a non-ASCII address, "NON-ASCII-LOCAL@example.com". Its display-name is "DISPLAY-LOCAL".

- o The "To:" header field contains two non-ASCII addresses, "NON-ASCII-REMOTE1@example.net" and "NON-ASCII-REMOTE2@example.com". Its display-names are "DISPLAY-REMOTE1" and "DISPLAY-REMOTE2".
- o The "Cc:" header field contains a non-ASCII address, "NON-ASCII-REMOTE3@example.org". Its display-name is "DISPLAY-REMOTE3".
- o Four display names contain non-ASCII characters.
- o The Subject header field is "NON-ASCII-SUBJECT", which contains non-ASCII strings.
- o The "Message-Id:" header field contains "NON-ASCII-MESSAGE\_ID", which contains non-ASCII strings.
- o There is an unknown header field "X-Unknown-Header" which contains non-ASCII strings.

```
Return-Path: <NON-ASCII-LOCAL@example.com>
Received: from ... by ... for <NON-ASCII-REMOTE1@example.net>
Received: from ... by ... for <NON-ASCII-REMOTE1@example.net>
From: DISPLAY-LOCAL <NON-ASCII-LOCAL@example.com>
To: DISPLAY-REMOTE1 <NON-ASCII-REMOTE1@example.net>,
    DISPLAY-REMOTE2 <NON-ASCII-REMOTE2@example.com>
Cc: DISPLAY-REMOTE3 <NON-ASCII-REMOTE3@example.org>
Subject: NON-ASCII-SUBJECT
Date: Mon, 30 Jul 2012 01:23:45 -0000
Message-Id: NON-ASCII-MESSAGE_ID
Mime-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
X-Unknown-Header: NON-ASCII-CHARACTERS
```

MAIL\_BODY

Figure 1: Received message in a mail drop

The downgraded message is shown in Figure 2. "Return-Path:", "From:", "To:" and "Cc:" header fields are rewritten. "Subject:" and "X-Unknown-Header:" header fields are encoded using [RFC2047]. "Message-Id:" header field is encapsulated as "Downgraded-Message-Id:" header field.

```
Return-Path: =?UTF-8?Q?NON-ASCII-LOCAL@example.com?= ;;
Received: from ... by ...
Received: from ... by ...
From: =?UTF-8?Q?DISPLAY-LOCAL?=
      =?UTF-8?Q?NON-ASCII-LOCAL@example.com?= ;;
To: =?UTF-8?Q?DISPLAY-REMOTE1?=
    =?UTF-8?Q?NON-ASCII-REMOTE1@example.net?= ;;,
    =?UTF-8?Q?DISPLAY-REMOTE2?=
    =?UTF-8?Q?NON-ASCII-REMOTE2@example.com?= ;;,
Cc: =?UTF-8?Q?DISPLAY-REMOTE3?=
    =?UTF-8?Q?NON-ASCII-REMOTE3@example.org?= ;;
Subject: =?UTF-8?Q?NON-ASCII-SUBJECT?=
Date: Mon, 30 Jul 2012 01:23:45 -0000
Downgraded-Message-Id: =?UTF-8?Q?MESSAGE_ID?=
Mime-Version: 1.0
Content-Type: text/plain; charset="UTF-8"
Content-Transfer-Encoding: 8bit
X-Unknown-Header: =?UTF-8?Q?NON-ASCII-CHARACTERS?=
```

MAIL\_BODY

Figure 2: Downgraded message

## Appendix B. Change History

[[RFC Editor: Please remove this section prior to publication.]]

This section is used for tracking the update of this document. Will be removed after finalize.

### B.1. Version 00

- o Initial version
- o Imported header field downgrading from RFC 5504

### B.2. Version 01

- o same as Version 00

### B.3. Version 02

- o Added updating RFC 5322 to allow <group> syntax in From: and Sender
- o Added GROUP Downgrading

## B.4. Version 03

- o Replaced <utf8-addr-spec> with <addr-spec>
- o Added updating RFC 5322 to allow <group> syntax in From: and Sender
- o Added one sentence in Security considerations
- o Updated IANA considerations

## B.5. Version 04

- o Removed "Internationalized Address removed" from GROUP and MAILBOX downgrading
- o Updated "Updating RFC 5322"
- o Compacted new header field definition
- o Compacted security considerations
- o Updated IANA considerations to remove obsoleting header fields that are registered by RFC 5504
- o Added a discussion of alternate downgrading models for the POP and IMAP cases.
- o Incorporated a large number of editorial changes to improve clarity.

## B.6. Version 05

- o Some text corrections
- o Terminology change: only to use non-ASCII address, non-ASCII message, non-ASCII string and imported them from RFC 6530 and RFC 6532
- o Replace "non-ASCII character" with "non-ASCII string"
- o Removed 5.1.1. RECEIVED Downgrading

## B.7. Version 06

- o Removed "Updating RFC 5322"

- o Added reference to draft-leiba-5322upd-from-group

#### B.8. Version 07

- o Updated by WGLC comments
- o Fixed Received downgrading and added to refer "RFC 6531", "RFC 5890", "RFC 5891"
- o Added Domain downgrading for Received, Group and Mailbox
- o Swapped section 3 and 4

#### B.9. Version 08

- o Updated by IETF Last call and IESG comments
- o Removed "Address Header Fields with Typed Addresses" and added "Delivery Status Notification downgrading" in MIME downgrading
- o Added a space between display-name and ENCODED\_WORD.
- o Moved "ENCAPSULATION: A Last Resort" from section 4 to section 3.1.10.
- o Updated address header fields downgrading
- o Updated introduction, security considerations and iana considerations

#### Author's Address

Kazunori Fujiwara  
Japan Registry Services Co., Ltd.  
Chiyoda First Bldg. East 13F, 3-8-1 Nishi-Kanda  
Chiyoda-ku, Tokyo 101-0065  
Japan

Phone: +81 3 5215 8451  
EMail: fujiwara@jprs.co.jp

