

Email Address Internationalization  
(EAI)  
Internet-Draft  
Intended status: Standards Track  
Expires: April 18, 2011

K. Fujiwara  
JPRS  
Oct 15, 2010

Post-delivery Message Downgrading for Internationalized Email Messages  
draft-ietf-eai-popimap-downgrade-00.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 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), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at  
<http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at  
<http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on April 18, 2011.

## Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	4
3. New Header Fields Definition . . . . .	4
3.1. Unknown Header Fields' Preservation Header Fields . . . . .	5
4. Email Header Fields Downgrading . . . . .	5
4.1. Downgrading Method for Each ABNF Element . . . . .	5
4.1.1. RECEIVED Downgrading . . . . .	6
4.1.2. UNSTRUCTURED Downgrading . . . . .	6
4.1.3. WORD Downgrading . . . . .	6
4.1.4. COMMENT Downgrading . . . . .	6
4.1.5. MIME-VALUE Downgrading . . . . .	6
4.1.6. DISPLAY-NAME Downgrading . . . . .	6
4.1.7. MAILBOX Downgrading . . . . .	6
4.1.8. ENCAPSULATION Downgrading . . . . .	7
4.1.9. TYPED-ADDRESS Downgrading . . . . .	7
4.2. Downgrading Method for Each Header Field . . . . .	7
4.2.1. Address Header Fields That Contain <address>s . . . . .	7
4.2.2. Address Header Fields with Typed Addresses . . . . .	8
4.2.3. Downgrading Non-ASCII in Comments . . . . .	8
4.2.4. Received Header Field . . . . .	9
4.2.5. MIME Content Header Fields . . . . .	9
4.2.6. Non-ASCII in <unstructured> . . . . .	9
4.2.7. Non-ASCII in <phrase> . . . . .	9
4.2.8. Other Header Fields . . . . .	9
5. MIME Body-Part Header Field Downgrading . . . . .	10
6. Security Considerations . . . . .	10
7. Implementation Notes . . . . .	11
7.1. RFC 2047 Encoding . . . . .	11
8. IANA Considerations . . . . .	11
9. Acknowledgements . . . . .	11
10. References . . . . .	12
10.1. Normative References . . . . .	12
10.2. Informative References . . . . .	13
Appendix A. Examples . . . . .	13
A.1. Downgrading Example . . . . .	13

## 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-frmrk-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 converts mail header fields and MIME header fields to ASCII.

This document does not change any protocols except by defining new header fields. It describes the conversion method from the internationalized email messages that are defined in [I-D.ietf-eai-frmrk-4952bis], and [I-D.ietf-eai-rfc5335bis] to the traditional email messages defined in [RFC5322].

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 three parts:

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

In Section 3 of this document, header fields starting with "Downgraded-" are introduced. They preserve the original header fields.

Email header field downgrading is described in Section 4. 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 5. 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-frmrwk-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-frmrwk-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. 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 or rewritten. 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.

### 3.1. Unknown Header Fields' Preservation Header Fields

The unknown header fields' preservation header fields are defined to encapsulate those original header fields that contain non-ASCII characters and are not otherwise provided for in this specification. The encapsulation header field name is the concatenation of "Downgraded-" and the original name. The value field holds the original header field value.

The header field syntax is specified as follows:

```
fields      =/ unknown-downgraded-headers ":" unstructured CRLF
unknown-downgraded-headers = "Downgraded-" original-header-field-name
original-header-field-name = field-name
field-name  = 1*ftext
ftext       = %d33-57 /           ; Any character except
              %d59-126           ; controls, SP, and ":".
```

To encapsulate 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 the result is ASCII.
3. Remove the original header field.

### 4. 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>.

#### 4.1. Downgrading Method for Each ABNF Element

Header field downgrading is defined below for each ABNF element.  
Downgrading an unknown header field is also defined as ENCAPSULATION

downgrading. Converting the header field terminates when no non-ASCII characters remain in the header field.

#### 4.1.1. 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.

#### 4.1.2. UNSTRUCTURED Downgrading

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

#### 4.1.3. WORD Downgrading

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

#### 4.1.4. COMMENT Downgrading

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

#### 4.1.5. 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.

#### 4.1.6. 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.

#### 4.1.7. 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, rewrite each <mailbox> element to ASCII-only format. The <mailbox> element that contains non-ASCII

characters is one of two formats.

```
[ Display-name ] "<" Utf8-addr-spec ">"
```

Utf8-addr-spec

Rewrite both as:

```
[ Display-name ] "Internationalized Address " Encoded-word  
" Removed:;"
```

where the <Encoded-word> is the original <Utf8-addr-spec> encoded according to [RFC2047].

[[ Note: If the original non-ASCII address is a part of a group address, this rewriting may conflict the original DISPLAY-NAME. This problem need to be fixed. ]]

#### 4.1.8. ENCAPSULATION Downgrading

If the header field contains non-ASCII characters and is such that no rule is given above, encapsulate it in a "Downgraded-" header field as described in Section 3.1 as a last resort.

Applying this procedure to "Received:" header field is prohibited.

#### 4.1.9. 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 [RFC5337]. 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.

#### 4.2. 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.

##### 4.2.1. 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 <mailbox> elements that contain non-ASCII addresses, perform COMMENT downgrading, DISPLAY-NAME downgrading, and MAILBOX downgrading.

[[ Note: RFC 5322 does not allow group syntax in "From:", "Resent-From:", "Sender:", "Resent-Sender:", but proposed method uses group syntax. This problem need to be fixed. ]]

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

Original-Recipient:  
Final-Recipient:

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

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

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

#### 4.2.4. Received Header Field

Received:

Perform COMMENT downgrading and RECEIVED downgrading.

#### 4.2.5. MIME Content Header Fields

Content-Type:  
Content-Disposition:

Perform MIME-VALUE downgrading and COMMENT downgrading.

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

Subject:  
Comments:  
Content-Description:

Perform UNSTRUCTURED downgrading.

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

Keywords:

Perform WORD downgrading.

#### 4.2.8. Other Header Fields

For all other header fields that contain non-ASCII characters, are user-defined, and are missing from this document or future defined header fields, perform ENCAPSULATION downgrading.

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

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

this category.

## 5. 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 4.

### 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.

## 6. Security Considerations

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-frmwk-4952bis] for more discussion.

## 7. Implementation Notes

### 7.1. 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.

## 8. IANA Considerations

IANA is requested to refuse registration of all field names that start with "Downgraded-". For unknown header fields, use the downgrading method described in Section 3.1 to avoid conflicts with existing IETF activity (Email Address Internationalization).

## 9. 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.

## 10. References

## 10.1. Normative References

- [I-D.ietf-eai-frmwrk-4952bis] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", draft-ietf-eai-frmwrk-4952bis-10 (work in progress), September 2010.
- [I-D.ietf-eai-rfc5335bis] Yang, A. and S. Steele, "Internationalized Email Headers", draft-ietf-eai-rfc5335bis-02 (work in progress), August 2010.
- [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.
- [RFC2979] Freed, N., "Behavior of and Requirements for Internet Firewalls", RFC 2979, October 2000.
- [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.
- [RFC5337] Newman, C. and A. Melnikov, "Internationalized Delivery Status and Disposition Notifications", RFC 5337, September 2008.

## 10.2. 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-remotel@example.net" and "NON-ASCII-remote2@example.com" Its display-names are "DISPLAY-remotel" 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 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-remotel@example.net>
Received: from ... by ... for <NON-ASCII-remotel@example.net>
From: DISPLAY-local <NON-ASCII-local@example.com>
To: DISPLAY-remotel <NON-ASCII-remotel@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: 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. "X-Unknown-Header:" is encapsulated as "Downgraded-X-Unknown-Header:".

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  
Message-Id: MESSAGE\_ID  
Mime-Version: 1.0  
Content-Type: text/plain; charset="UTF-8"  
Content-Transfer-Encoding: 8bit  
Downgraded-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@jprs.co.jp





Email Address Internationalization  
(EAI)  
Internet-Draft  
Obsoletes: 5335 (if approved)  
Updates: 2045,5321,5322  
(if approved)  
Intended status: Standards Track  
Expires: April 25, 2011

A. Yang  
TWNIC  
S. Steele  
Microsoft  
October 22, 2010

Internationalized Email Headers  
draft-ietf-eai-rfc5335bis-03

Abstract

Full internationalization of electronic mail requires not only the capabilities to transmit non-ASCII content, to encode selected information in specific header fields, and to use non-ASCII characters in envelope addresses. It also requires being able to express those addresses and the information based on them in mail header fields. This document specifies a variant of Internet mail that permits the use of Unicode encoded in UTF-8, rather than ASCII, as the base form for Internet email header field. This form is permitted in transmission only if authorized by an SMTP extension, as specified in an associated specification. This specification updates Section 6.4 of [RFC2045] to conform with the requirements.

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, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
1.1. Role of This Specification . . . . .	3
1.2. Relation to Other Standards . . . . .	3
2. Background and History . . . . .	3
3. Terminology . . . . .	4
4. Changes on Message Header Fields . . . . .	5
4.1. UTF-8 Syntax and Normalization . . . . .	5
4.2. Changes on MIME Headers . . . . .	6
4.3. Syntax Extensions to RFC 5322 . . . . .	6
4.4. Change on addr-spec Syntax . . . . .	8
4.5. Trace Field Syntax . . . . .	9
4.6. message/global . . . . .	9
5. Security Considerations . . . . .	11
6. IANA Considerations . . . . .	12
7. Acknowledgements . . . . .	12
8. Edit history . . . . .	12
8.1. draft-ietf-eai-rfc5335bis-00 . . . . .	12
8.2. draft-ietf-eai-rfc5335bis-01 . . . . .	12
8.3. draft-ietf-eai-rfc5335bis-02 . . . . .	13
8.4. draft-ietf-eai-rfc5335bis-03 . . . . .	13
9. References . . . . .	13
9.1. Normative References . . . . .	13
9.2. Informative References . . . . .	14

## 1. Introduction

### 1.1. Role of This Specification

Full internationalization of electronic mail requires several capabilities:

- o The capability to transmit non-ASCII content, provided for as part of the basic MIME specification [RFC2045], [RFC2046].
- o The capability to use international characters in envelope addresses, discussed in [I-D.ietf-eai-frmrwk-4952bis] and specified in [I-D.ietf-eai-rfc5336bis].
- o The capability to express those addresses, and information related to them and based on them, in mail header fields, defined in this document.

This document specifies a variant of Internet mail that permits the use of Unicode encoded in UTF-8 [RFC3629], rather than ASCII, as the base form for Internet email header fields. This form is permitted in transmission, if authorized by the SMTP extension specified in [I-D.ietf-eai-rfc5336bis] or by other transport mechanisms capable of processing it.

### 1.2. Relation to Other Standards

This document updates Section 6.4 of [RFC2045]. It removes the blanket ban on applying a content-transfer-encoding to all subtypes of message/, and instead specifies that a composite subtype MAY specify whether or not a content-transfer-encoding can be used for that subtype, with "cannot be used" as the default.

This document also updates [RFC5322] and MIME ([RFC2045]).

Allowing use of a content-transfer-encoding on subtypes of messages is not limited to transmissions that are authorized by the SMTP extension specified in [I-D.ietf-eai-rfc5336bis]. message/global (see Section 4.6) permits use of a content-transfer-encoding.

## 2. Background and History

Mailbox names often represent the names of human users. Many of these users throughout the world have names that are not normally expressed with just the ASCII repertoire of characters, and would like to use more or less their real names in their mailbox names. These users are also likely to use non-ASCII text in their common names and subjects of email messages, both received and sent. This

protocol specifies UTF-8 as the encoding to represent email header field bodies.

The traditional format of email messages [RFC5322] allows only ASCII characters in the header fields of messages. This prevents users from having email addresses that contain non-ASCII characters. It further forces non-ASCII text in common names, comments, and in free text (such as in the "Subject:" field) to be encoded (as required by MIME format [RFC2047]). This specification describes a change to the email message format that is related to the SMTP message transport change described in the associated documents [I-D.ietf-eai-frmrk-4952bis] and [I-D.ietf-eai-rfc5336bis], and that allows non-ASCII characters in most email header fields. These changes affect SMTP clients, SMTP servers, mail user agents (MUAs), list expanders, gateways to other media, and all other processes that parse or handle email messages.

As specified in [I-D.ietf-eai-rfc5336bis], an SMTP protocol extension "UTF8SMTPbis" is used to prevent the transmission of messages with UTF-8 header fields to systems that cannot handle such messages. [[Note in Draft: Keyword related to UTF8SMTP will be decided by WG before publication.]]

Use of this SMTP extension helps prevent the introduction of such messages into message stores that might misinterpret, improperly display, or mangle such messages. It should be noted that using an ESMTP extension does not prevent transferring email messages with UTF-8 header fields to other systems that use the email format for messages and that may not be upgraded, such as unextended POP and IMAP servers. Changes to these protocols to handle UTF-8 header fields are addressed in [RFC5721bis] and [RFC5738bis]. [[Note in Draft: RFC5721bis and RFC5738bis did not yet posted.]]

The objective for this protocol is to allow UTF-8 in email header fields.

### 3. Terminology

A plain ASCII string is also a valid UTF-8 string; see [RFC3629]. In this document, ordinary ASCII characters are UTF-8 characters if they are in headers which contain <utf8-xtra-char>s.

Unless otherwise noted, all terms used here are defined in [RFC5321], [RFC5322], [I-D.ietf-eai-frmrk-4952bis], or [I-D.ietf-eai-rfc5336bis].

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this

document are to be interpreted as described in [RFC2119].

#### 4. Changes on Message Header Fields

SMTP clients can send header fields in UTF-8 format, if the UTF8SMTPbis extension is advertised by the SMTP server or is permitted by other transport mechanisms.

This protocol does NOT change the [RFC5322] rules for defining header field names. The bodies of header fields are allowed to contain UTF-8 characters, but the header field names themselves must contain only ASCII characters.

To permit UTF-8 characters in field values, the header definition in [RFC5322] is extended to support the new format. The following ABNF is defined to substitute those definitions in [RFC5322].

The syntax rules not covered in this section remain as defined in [RFC5322].

##### 4.1. UTF-8 Syntax and Normalization

UTF-8 characters can be defined in terms of octets using the following ABNF [RFC5234], taken from [RFC3629]:

UTF8-xtra-char= UTF8-2 / UTF8-3 / UTF8-4

UTF8-2 = %xC2-DF UTF8-tail

UTF8-3 = %xE0 %xA0-BF UTF8-tail / %xE1-EC 2(UTF8-tail) /  
%xED %x80-9F UTF8-tail / %xEE-EF 2(UTF8-tail)

UTF8-4 = %xF0 %x90-BF 2( UTF8-tail ) / %xF1-F3 3( UTF8-tail ) /  
%xF4 %x80-8F 2( UTF8-tail )

UTF8-tail = %x80-BF

These are normatively defined in [RFC3629], but kept in this document for reasons of convenience.

See [RFC5198] for a discussion of normalization; the use of normalization form NFC is RECOMMENDED. Actually, if one is going to do internationalization properly, one of the most often-cited goals is to permit people to spell their names correctly. Since many mailbox local parts reflect personal names, that principle applies as well. And NFKC is not recommended because it may lose information that is needed to correctly spell some names except in unusual circumstances.

#### 4.2. Changes on MIME Headers

This specification updates Section 6.4 of [RFC2045]. [RFC2045] prohibits applying a content-transfer-encoding to all subtypes of message/. This specification relaxes the rule -- it allows newly defined MIME types to permit content-transfer-encoding, and it allows content-transfer-encoding for message/global (see Section 4.6).

Background: Normally, transfer of message/global will be done in 8-bit-clean channels, and body parts will have "identity" encodings, that is, no decoding is necessary. In the case where a message containing a message/global is downgraded from 8-bit to 7-bit as described in [RFC1652], an encoding may be applied to the message; if the message travels multiple times between a 7-bit environment and an environment implementing UTF8SMTPbis, multiple levels of encoding may occur. This is expected to be rarely seen in practice, and the potential complexity of other ways of dealing with the issue are thought to be larger than the complexity of allowing nested encodings where necessary.

#### 4.3. Syntax Extensions to RFC 5322

The following rules are intended to extend the corresponding rules in [RFC5322] in order to allow UTF-8 characters.

FWS = <see [RFC5322] Section 3.2.2, folding white space>

CFWS = <see [RFC5322] Section 3.2.2>

ctext =/ UTF8-xtra-char

utext =/ UTF8-xtra-char

comment = "(" \*([FWS] uCcontent) [FWS] ")"

word = uAtom / uQuoted-String

This means that all the [RFC5322] constructs that build upon these will permit UTF-8 characters, including comments and quoted strings. We do not change the syntax of <atext> in order to allow UTF-8 characters in <addr-spec>. This would also allow UTF-8 characters in <message-id>, which is not allowed due to the limitation described in Section 4.5. Instead, <uAtext> is added to meet this requirement.

```

uText      = %d1-9 /      ; all UTF-8 characters except
               %d11-12 /   ; US-ASCII NUL, CR, and LF
               %d14-127 /
               UTF8-xtra-char

uQuoted-Pair = ("\" uText) / obs-qp

uQcontent   = uQtext / uQuoted-Pair

uQuoted-String = [CFWS] DQUOTE *([FWS] uQcontent) [FWS] DQUOTE [CFWS]

DQUOTE      = <See appendix B.1 of RFC 5234>

uCcontent   = ctext / uQuoted-Pair / comment

uQtext      = qtext / UTF8-xtra-char

uAtext      = ALPHA / DIGIT /
               "!" / "#" / ; Any character except
               "$" / "%" / ; controls, SP, and specials.
               "&" / "'" / ; Used for atoms.
               "*" / "+" /
               "-" / "/" /
               "=" / "?" /
               "^" / "_" /
               "`" / "{" /
               "|" / "}" /
               "~" /
               UTF8-xtra-char

uAtom       = FWS] 1*uAtext [CFWS]

uDot-Atom   = CFWS] uDot-Atom-text [CFWS]

uDot-Atom-text = 1*uAtext *("." 1*uAtext)

qcontent    = uQcontent

```

To allow the use of UTF-8 in a Content-Description header field [RFC2045], the following syntax is used:

```
description = "Content-Description:" unstructured CRLF
```

The <utext> syntax is extended above to allow UTF-8 in all <unstructured> header fields.

Note, however, this does not remove any constraint on the character

set of protocol elements; for instance, all the allowed values for timezone in the "Date:" headers are still expressed in ASCII. And also, none of this revised syntax changes what is allowed in a <message-id>, which will still remain in pure ASCII.

#### 4.4. Change on addr-spec Syntax

Internationalized email addresses are represented in UTF-8. Thus, all header fields containing <mailbox>es are updated from [RFC5321] Section 4.1.2 to permit UTF-8 addresses.

```

mailbox      = name-addr / addr-spec / uAddr-Spec
               ; Replace mailbox in Section 3.4.1 of RFC 5322
angle-addr   =/ [CFWS] "<" uAddr-Spec">" [CFWS] / obs-angle-addr
               ; Replace angle-addr in Section 3.4 of RFC 5322

uAddr-Spec   = uLocal-Part "@" uDomain

uLocal-Part  = uDot-String / uQuoted-String
               ; Replace Local-Part in Section 3.4.1 of RFC 5322

uDot-string  = uAtom *("." uAtom)

uDomain      = (sub-udomain 1*("." sub-uDomain)) /
               dot-atom / domain-literal / obs-domain
               ; Replace Domain in Section 4.1.2 of RFC 5321

sub-udomain  = uLet-dig [uLdh-str]
               ; Replace sub-domain in Section 4.1.2 of RFC 5321

uLet-dig     = Let-dig / UTF8-xtra-char
               ; Replace uLet-dig in Section 4.1.2 of RFC 5321

Let-dig      = <See Section 4.1.3 of RFC 5321>

uLdh-str     = *( ALPHA / DIGIT / "-" / UTF8-xtra-char ) uLet-dig
               ; Replace Ldh-str Section 4.1.2 of RFC 5321

```

Below are a few examples of possible <mailbox> representations.



```
"DISPLAY_NAME" <ASCII@ASCII>
; traditional mailbox format
"DISPLAY_NAME" <non-ASCII@non-ASCII>
; message will bounce if UTF8SMTPbis extension is not supported
<non-ASCII@non-ASCII>
; without DISPLAY_NAME and quoted string
; message will bounce if UTF8SMTPbis extension is not supported
```

#### 4.5. Trace Field Syntax

The uFor ( described in [I-D.ietf-eai-rfc5336bis] Section 3.6.3 )) has been added to allow the use of internationalized addresses in "For" fields. by use of the new uFor syntax. UTF-8 information may be needed in "Received:" fields. Such information is therefore allowed to preserve the integrity of those fields. The uFor syntax retains the original UTF-8 email address between email address internationalization EAI-aware MTAs.

The "Return-path" designates the address to which messages indicating non-delivery or other mail system failures are to be sent. Thus, the header is augmented to carry UTF-8 addresses (see the revised syntax of <angle-addr> in Section 4.4 of this document). This will not break the rule of trace field integrity, because the header field is added at the last MTA and described in [RFC5321].

The <item-value> on "Received:" field syntax is augmented to allow UTF-8 email address in the "For" field. <angle-addr> is augmented to include UTF-8 email address. In order to allow UTF-8 email addresses in an <addr-spec>, <uAddr-Spec> is added to <item-value>.

```
item-value      =/      uAddr-Spec
```

#### 4.6. message/global

Internationalized messages MUST only be transmitted as authorized by [I-D.ietf-eai-rfc5336bis] or within a non-SMTP environment which supports these messages. A message is a "message/global message", if

- o it contains UTF-8 header values as specified in this document, or
- o it contains UTF-8 values in the headers fields of body parts.

The type message/global is similar to message/rfc822, except that it specifies that a message can contain UTF-8 characters in the headers of the message or body parts. If this type is sent to a 7-bit-only

system, it has to be encoded in MIME [RFC2045]. (Note that a system compliant with MIME that doesn't recognize message/global MUST treat it as "application/octet-stream" as described in Section 5.2.4 of [RFC2046].)

Type name: message

Subtype name: global

Required parameters: none

Optional parameters: none

Encoding considerations: Any content-transfer-encoding is permitted. The 8-bit or binary content-transfer-encodings are recommended where permitted.

Security considerations: See Section 5.

Interoperability considerations: The media type provides functionality similar to the message/rfc822 content type for email messages with international email headers. When there is a need to embed or return such content in another message, there is generally an option to use this media type and leave the content unchanged or down-convert the content to message/rfc822. Both of these choices will interoperate with the installed base, but with different properties. Systems unaware of internationalized headers will typically treat a message/global body part as an unknown attachment, while they will understand the structure of a message/rfc822. However, systems that understand message/global will provide functionality superior to the result of a down-conversion to message/rfc822. The most interoperable choice depends on the deployed software.

Published specification: RFC XXXX

Applications that use this media type: SMTP servers and email clients that support multipart/report generation or parsing. Email clients which forward messages with international headers as attachments.

Additional information:

Magic number(s): none

File extension(s): The extension ".u8msg" is suggested.

Macintosh file type code(s): A uniform type identifier (UTI) of "public.utf8-email-message" is suggested. This conforms to "public.message" and "public.composite-content", but does not necessarily conform to "public.utf8-plain-text".

Person & email address to contact for further information: See the Author's Address section of this document.

Intended usage: COMMON

Restrictions on usage: This is a structured media type which embeds other MIME media types. The 8-bit or binary content-transfer-encoding MUST be used unless this media type is sent over a 7-bit-only transport.

Author: See the Author's Address section of this document.

Change controller: IETF Standards Process

## 5. Security Considerations

If a user has a non-ASCII mailbox address and an ASCII mailbox address, a digital certificate that identifies that user may have both addresses in the identity. Having multiple email addresses as identities in a single certificate is already supported in PKIX (Public Key Infrastructure for X.509 Certificates) and OpenPGP.

Because UTF-8 often requires several octets to encode a single character, internationalized local parts may cause mail addresses to become longer. As specified in [RFC5322], each line of characters MUST be no more 998 octets, excluding the CRLF.

Because internationalized local parts may cause email addresses to be longer, processes that parse, store, or handle email addresses or local parts must take extra care not to overflow buffers, truncate addresses, or exceed storage allotments. Also, they must take care, when comparing, to use the entire lengths of the addresses.

In this specification, a user could provide an ASCII alternative address for a non-ASCII address. However, it is possible these two addresses go to different mailboxes, or even different people. This configuration may be based on a user's personal choice or on administration policy. We recognize that if ASCII and non-ASCII email is delivered to two different destinations, based on MTA capability, this may violate the principle of least astonishment, but this is not a "protocol problem".

The security impact of UTF-8 headers on email signature systems such as Domain Keys Identified Mail (DKIM), S/MIME, and OpenPGP is discussed in [I-D.ietf-eai-frmrwk-4952bis], Section 14.

## 6. IANA Considerations

IANA has registered the message/global MIME type using the registration form contained in Section 4.4.

## 7. Acknowledgements

This document incorporates many ideas first described in Internet-Draft form by Paul Hoffman, although many details have changed from that earlier work.

The author especially thanks Jeff Yeh for his efforts and contributions on editing previous versions.

Most of the content of this document is provided by John C Klensin. Also, some significant comments and suggestions were received from Charles H. Lindsey, Kari Hurtta, Pete Resnick, Alexey Melnikov, Chris Newman, Yangwoo Ko, Yoshiro Yoneya, and other members of the JET team (Joint Engineering Team) and were incorporated into the document. The editor sincerely thanks them for their contributions.

## 8. Edit history

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

### 8.1. draft-ietf-eai-rfc5335bis-00

1. Applied Errata suggested by Alfred Hoenes.
2. Adjust [RFC2821] and [RFC2822] to [RFC5321] and [RFC5322].
3. Abrogate <alt-address> in ABNF of <angle-addr>.
4. Revoke [RFC5504] from this document.
5. Upgrade some references from I-Ds to RFC.

### 8.2. draft-ietf-eai-rfc5335bis-01

1. Author name revised.

## 8.3. draft-ietf-eai-rfc5335bis-02

1. ABNF revised.

## 8.4. draft-ietf-eai-rfc5335bis-03

1. Fix typos
2. ABNF revised
3. Improve sentence

## 9. References

## 9.1. Normative References

- [I-D.ietf-eai-frmwrk-4952bis] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", draft-ietf-eai-frmwrk-4952bis-10 (work in progress), September 2010.
- [I-D.ietf-eai-rfc5336bis] Yao, J. and W. MAO, "SMTP Extension for Internationalized Email Address", draft-ietf-eai-rfc5336bis-04 (work in progress), October 2010.
- [RFC1652] Klensin, J., Freed, N., Rose, M., Stefferud, E., and D. Crocker, "SMTP Service Extension for 8bit-MIMEtransport", RFC 1652, July 1994.
- [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.
- [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.

- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.

## 9.2. Informative References

- [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.

## Authors' Addresses

Abel Yang  
TWNIC  
4F-2, No. 9, Sec 2, Roosevelt Rd.  
Taipei, 100  
Taiwan

Phone: +886 2 23411313 ext 505  
EMail: abelyang@twNIC.net.tw

Shawn Steele  
Microsoft

EMail: Shawn.Steele@microsoft.com



Network Working Group  
Internet-Draft  
Obsoletes: RFC5336  
(if approved)  
Updates: RFC5321 and 5322  
(if approved)  
Intended status: Standards Track  
Expires: April 24, 2011

Jiankang. Yao  
Wei. Mao  
CNNIC  
October 21, 2010

SMTP Extension for Internationalized Email Address  
draft-ietf-eai-rfc5336bis-04.txt

Abstract

This document specifies an SMTP extension for transport and delivery of email messages with internationalized email addresses or header information. This document updates some syntaxes and rules defined in RFC 5321 and RFC 5322.

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 24, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must



include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF 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
1.1. Role of This Specification . . . . .	3
1.2. Terminology . . . . .	3
2. Overview of Operation . . . . .	4
3. Mail Transport-Level Protocol . . . . .	4
3.1. Framework for the Internationalization Extension . . . . .	4
3.2. The UTF8SMTPbis Extension . . . . .	5
3.3. Extended Mailbox Address Syntax . . . . .	6
3.4. UTF8 addresses and Response Codes . . . . .	7
3.5. Body Parts and SMTP Extensions . . . . .	7
3.6. Additional ESMTP Changes and Clarifications . . . . .	8
3.6.1. The Initial SMTP Exchange . . . . .	8
3.6.2. Mail eXchangers . . . . .	8
3.6.3. Trace Information . . . . .	9
3.6.4. UTF-8 Strings in Replies . . . . .	10
4. IANA Considerations . . . . .	11
5. Security Considerations . . . . .	13
6. Acknowledgements . . . . .	13
7. Change History . . . . .	14
7.1. draft-yao-eai-rfc5336bis: Version 00 . . . . .	14
7.2. draft-ietf-eai-rfc5336bis: Version 00 . . . . .	14
7.3. draft-ietf-eai-rfc5336bis: Version 01 . . . . .	14
7.4. draft-ietf-eai-rfc5336bis: Version 02 . . . . .	14
7.5. draft-ietf-eai-rfc5336bis: Version 03 . . . . .	14
7.6. draft-ietf-eai-rfc5336bis: Version 04 . . . . .	14
8. References . . . . .	14
8.1. Normative References . . . . .	14
8.2. Informative References . . . . .	16
Authors' Addresses . . . . .	16

## 1. Introduction

An internationalized email address includes two parts, the local part and the domain part. The ways email addresses are used by protocols are different from the ways domain names are used. The most critical difference is that emails are delivered through a chain of clients and servers, while domain names are resolved by name servers looking up those names in their own tables. The Simple Mail Transfer Protocol [RFC5321] provides a negotiation mechanism about service extension with which clients can discover server capabilities and make decisions for further processing. An extended overview of the extension model for internationalized addresses and headers appears in [RFC4952bis], referred to as "the framework document" or just as "framework" elsewhere in this specification. This document specifies an SMTP extension to permit internationalized email addresses in envelopes, and UNICODE characters (encoded in UTF-8) [RFC3629] in headers.

### 1.1. Role of This Specification

The framework document specifies the requirements for, and describes components of, full internationalization of the electronic mail. A thorough understanding of the information in that document and in the base Internet email specifications [RFC5321] [RFC5322] is necessary to understand and implement this specification.

This document specifies an element of the email internationalization work, specifically the definition of an SMTP extension [RFC5321] for internationalized email address transport delivery.

### 1.2. 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].

The terms "UTF-8 string" or "UTF-8 character" are used informally to refer to Unicode characters encoded in UTF-8 [RFC3629]. All other specialized terms used in this specification are defined in the framework document [RFC4952bis] or in the base Internet email specifications [RFC5321] [RFC5322]. In particular, the terms "ASCII address", "internationalized email address", "non-ASCII address", "i18n email address", "UTF8SMTPbis", "conventional message", "internationalized message", "message", and "mailing list" are used in this document according to the definitions in the framework document [RFC4952bis].

This specification defines only those Augmented BNF (ABNF) [RFC5234] syntax rules that are different from those of the base email specifications [RFC5321][RFC5322] and, where the earlier rules are upgraded or extended, gives them new names. When the new rule is a small modification to the older one, it is typically given a name starting with "u". Rules that are undefined here may be found in the base email specifications under the same names.

## 2. Overview of Operation

This specification describes an optional extension to the email transport mechanism that permits non-ASCII [ASCII] characters in both the envelope and header fields of messages, which are encoded with UTF-8 [RFC3629] characters. The extension is identified with the token "UTF8SMTPbis". In order to provide information that may be needed in downgrading, an optional alternate ASCII address may be needed if an SMTP client attempts to transfer an internationalized message and encounters a server that does not support this extension.

The EAI UTF-8 header specification [RFC5335bis] provides the details of how and where non-ASCII characters are permitted in the header fields of messages. The context for this specification is described in the framework document.

## 3. Mail Transport-Level Protocol

### 3.1. Framework for the Internationalization Extension

The following service extension is defined:

1. The name of the SMTP service extension is "Email Address Internationalization".
2. The EHLO keyword value associated with this extension is "UTF8SMTPbis".
3. No parameter values are defined for this EHLO keyword value. In order to permit future (although unanticipated) extensions, the EHLO response MUST NOT contain any parameters for that keyword. Clients MUST ignore any parameters; that is, clients MUST behave as if the parameters do not appear. If a server includes UTF8SMTPbis in its EHLO response, it MUST be fully compliant with this version of this specification.
4. One optional parameter "UTF8REPLY" is added to the VRFY and EXPN commands. The parameter UTF8REPLY has no value. The parameter indicates that the SMTP client can accept Unicode characters in UTF-8 encoding in replies from the VRFY and EXPN commands.

5. No additional SMTP verbs are defined by this extension.
6. Servers offering this extension MUST provide support for, and announce, the 8BITMIME extension [RFC1652].
7. The reverse-path and forward-path of the SMTP MAIL and RCPT commands are extended to allow Unicode characters encoded in UTF-8 in mailbox names (addresses).
8. The mail message body is extended as specified in [RFC5335bis].
9. The UTF8SMTPbis extension is valid on the submission port [RFC4409].

### 3.2. The UTF8SMTPbis Extension

An SMTP server that announces this extension MUST be prepared to accept a UTF-8 string [RFC3629] in any position in which RFC 5321 specifies that a mailbox can appear. That string MUST be parsed only as specified in [RFC5321], i.e., by separating the mailbox into source route, local part, and domain part, using only the characters colon (U+003A), comma (U+002C), and at-sign (U+0040) as specified there. Once isolated by this parsing process, the local part MUST be treated as opaque unless the SMTP server is the final delivery Mail Transfer Agent (MTA). Any domain names to be looked up in the DNS MUST allow for [RFC5890] behavior. When doing lookups, the server MUST either use a Unicode aware DNS library, or transform it to A-label defined in [RFC5890]. Any domain names that are to be compared to local strings SHOULD be checked for validity and then MUST be compared as specified in section 3 of [RFC5891].

An SMTP client that receives the UTF8SMTPbis extension keyword in response to the EHLO command MAY transmit mailbox names within SMTP commands as internationalized strings in UTF-8 form. It MAY send a UTF-8 header [RFC5335bis] (which may also include mailbox names in UTF-8). It MAY transmit the domain parts of mailbox names within SMTP commands or the message header as either ACE (ASCII Compatible Encoding) labels (as specified in IDNA definitions [RFC5890]) or UTF-8 strings. All labels in domain parts of mailbox names which are IDNs (either UTF-8 or ACE strings) MUST be valid. When a Message User Agent submits a message to a Message Submission Server ("MSA") [RFC4409], it is the responsibility of the MSA to ensure that all domain labels are valid. The presence of the UTF8SMTPbis extension does not change the requirement of RFC 5321 that servers relaying mail MUST NOT attempt to parse, evaluate, or transform the local part in any way.

If the UTF8SMTPbis SMTP extension is not offered by the server, the SMTP client MUST NOT transmit an internationalized address and MUST NOT transmit a mail message containing internationalized mail headers as described in [RFC5335bis] at any level within its MIME structure [RFC2045] and [RFC2047]. (For this paragraph, the internationalized

domain name in the form of ACE labels as specified in IDNA definitions [RFC5890] is not considered to be "internationalized".) Instead, if an SMTP client (SMTP sender) attempts to transfer an internationalized message and encounters a server that does not support the extension, it MUST make one of the following three choices:

1. If and only if the SMTP client (sender) is a Message Submission Server ("MSA") [RFC4409], it MAY, consistent with the general provisions for changes by such servers, rewrite the envelope, headers, or message material to make them entirely ASCII and consistent with the provisions of RFC 5321 [RFC5321] and RFC 5322 [RFC5322].
2. It may either reject the message during the SMTP transaction or accept the message and then generate and transmit a notification of non-deliverability. Such notification MUST be done as specified in RFC 5321 [RFC5321], RFC 3464 [RFC3464], and the EAI delivery status notification (DSN) specification [RFC5337bis].
3. It may find an alternate route to the destination that permits UTF8SMTPbis. That route may be discovered by trying alternate Mail eXchanger (MX) hosts (using preference rules as specified in RFC 5321) or using other means available to the SMTP-sender.

This document applies only when an UTF8SMTPbis-aware client is trying to send an internationalized message to a server which requires the UTF8SMTPbis extensions to handle it. For all other cases, and for addresses and messages that do not require an UTF8SMTPbis extension, SMTP clients and servers are expected to behave exactly as specified in [RFC5321].

A UTF8SMTPbis aware MUA/MSA sending to a legacy SMTP server [RFC5321] and [RFC5322] MAY convert the ASCII@non-ASCII address into the format of ASCII@A-label [RFC5890] if the email address is in the format of ASCII@non-ASCII.

### 3.3. Extended Mailbox Address Syntax

RFC 5321, Section 4.1.2, defines the syntax of a mailbox entirely in terms of ASCII characters, using the production for a mailbox and those productions on which it depends.

The key changes made by this specification are, informally, to

- o Change the definition of "Domain" to permit either the definition above or a UTF-8 string representing a DNS label that is conformant with IDNA definitions [RFC5890].
- o Change the definition of "Local-part" to permit either the definition above or a UTF-8 string. That string MUST NOT contain any of the ASCII characters (either graphics or controls) that are

not permitted in "atext"; it is otherwise unrestricted.

According to the description above, the syntax of an internationalized email mailbox name (address) is defined in ABNF [RFC5234] as follows.

```
uMailbox = uLocal-part "@" uDomain
          ; uLocal-part and uDomain defined
          ; in RFC 5335bis, Section 4.
```

The value of "uDomain" SHOULD be verified by IDNA definitions [RFC5890]. If that verification fails, the email address with that uDomain MUST NOT be regarded as a valid email address.

### 3.4. UTF8 addresses and Response Codes

An "internationalized message" as defined in the appendix of this specification MUST NOT be sent to an SMTP server that does not support UTF8SMTPbis. Such a message should be rejected by a server if it lacks the support of UTF8SMTPbis.

The three-digit reply codes used in this section are consistent with their meanings as defined in RFC 5321.

When messages are rejected because the RCPT command requires an ASCII address, the response code 553 is used with the meaning "mailbox name not allowed". When messages are rejected for other reasons, such as the MAIL command requiring an ASCII address, the response code 550 is used with the meaning "mailbox unavailable". When the server supports enhanced mail system status codes [RFC3463], response code "X.6.7" [RFC5248] is used, meaning that "UTF-8 addresses not permitted for that sender/recipient".

If the response code is issued after the final "." of the DATA command, the response code "554" is used with the meaning "Transaction failed". When the server supports enhanced mail system status codes [RFC3463], response code "X.6.9" [RFC5248] is used, meaning that "UTF-8 header message can not be transferred to one or more recipient so the message must be bounced".

### 3.5. Body Parts and SMTP Extensions

There is no ESMTP parameter to assert that a message is an internationalized message. An SMTP server that requires accurate knowledge of whether a message is internationalized is required to parse all message header fields and MIME header fields [RFC2045] and [RFC2047] in the message body.

While this specification requires that servers support the 8BITMIME extension [RFC1652] to ensure that servers have adequate handling capability for 8-bit data and to avoid a number of complex encoding problems, the use of internationalized addresses obviously does not require non-ASCII body parts in the MIME message [RFC2045] and [RFC2047]. The UTF8SMTPbis extension MAY be used with the BODY=8BITMIME parameter if that is appropriate given the body content or, with the BODY=BINARYMIME parameter, if the server advertises BINARYMIME [RFC3030] and that is appropriate.

Assuming that the server advertises UTF8SMTPbis and 8BITMIME, and receives at least one non-ASCII address, the precise interpretation of "BODY=8BITMIME", and "BODY=BINARYMIME" in the MAIL command is:

1. If a BODY=8BITMIME parameter is present, the header contains UTF-8 characters, and some or all of the body parts contain 8-bit line-oriented data.
2. If a BODY=BINARYMIME parameter is present, the header contains UTF-8 characters, and some or all body parts contain binary data without restriction as to line lengths or delimiters.

### 3.6. Additional ESMTP Changes and Clarifications

The information carried in the mail transport process involves addresses ("mailboxes") and domain names in various contexts in addition to the MAIL and RCPT commands and extended alternatives to them. In general, the rule is that, when RFC 5321 specifies a mailbox, this specification expects UTF-8 to be used for the entire string; when RFC 5321 specifies a domain name, the name SHOULD be in the form of ACE labels if its raw form is non-ASCII.

The following subsections list and discuss all of the relevant cases.

#### 3.6.1. The Initial SMTP Exchange

When an SMTP connection is opened, the server normally sends a "greeting" response consisting of the 220 response code and some information. The client then sends the EHLO command. Since the client cannot know whether the server supports UTF8SMTPbis until after it receives the response from EHLO, any domain names that appear in this dialogue, or in responses to EHLO, MUST be in the hostname form, i.e., internationalized ones MUST be in the form of ACE labels.

#### 3.6.2. Mail eXchangers

Organizations often authorize multiple servers to accept mail addressed to them. For example, the organization may itself operate more than one server, and may also or instead have an agreement with

other organizations to accept mail as a backup. Authorized servers are generally listed in MX records as described in RFC 5321. When more than one server accepts mail for the domain-part of a mailbox, it is strongly advised that either all or none of them support the UTF8SMTPbis extension. Otherwise, surprising downgrades can happen during temporary failures, which users might perceive as a serious reliability issue.

### 3.6.3. Trace Information

When an SMTP server receives a message for delivery or further processing, it MUST insert trace ("time stamp" or "Received") information at the beginning of the message content. "Time stamp" or "Received" appears in the form of "Received:" lines. The most important use of Received: lines is for debugging mail faults. When the delivery SMTP server makes the "final delivery" of a message, it inserts a Return-path line at the beginning of the mail data. The primary purpose of the Return-path is to designate the address to which messages indicating non-delivery or other mail system failures are to be sent. For the trace information, this memo updates the time stamp line and the return path line [RFC5321] formally defined as follows:

```
uReturn-path-line = "Return-Path:" FWS uReverse-path <CRLF>
    ; Replaces Return-path-line in Section 4.4 of RFC 5321
    ; uReverse-path is defined in Section 4 of RFC5335bis

uTime-stamp-line = "Received:" FWS uStamp <CRLF>
    ; Replaces Time-stamp-line in Section 4.4 of RFC 5321

uStamp = From-domain By-domain uOpt-info ";" FWS date-time
    ; Replaces Stamp in Section 4.4 of RFC 5321

uOpt-info = [Via] [With] [ID] [uFor]
    ; Replaces Opt-info in Section 4.4 of RFC 5321
    ; The protocol value for With will allow a UTF8SMTPbis value

uFor = "FOR" ( FWS (uPath / uMailbox) ) CFWS
    ; Replaces For in Section 4.4 of RFC 5321
    ; uMailbox is defined in section 3.3 of this document
uPath = "<" [ A-d-l ":" ] uMailbox ">"
    ; Replace Path in RFC 5321, section 4.1.2
    ; A-d-l is defined in RFC 5321, section 4.1.2
    ; uMailbox is defined in section 3.3 of this document
```

Except in the 'uFor' clause and 'uReverse-path' value where non-ASCII domain names may be used, internationalized domain names in Received fields MUST be transmitted in the form of ACE labels. The protocol



value of the WITH clause when this extension is used is one of the UTF8SMTPbis values specified in the "IANA Considerations" section of this document.

#### 3.6.4. UTF-8 Strings in Replies

##### 3.6.4.1. MAIL and RCPT Commands

If the client issues a RCPT command containing non-ASCII characters, the SMTP server is permitted to use UTF-8 characters in the email address associated with 251 and 551 response codes.

If an SMTP client follows this specification and sends any RCPT commands containing non-ASCII addresses, it MUST be able to accept and process 251 or 551 responses containing UTF-8 email addresses. If a given RCPT command does not include a non-ASCII envelope address, the server MUST NOT return a 251 or 551 response containing a non-ASCII mailbox. Instead, it MUST transform such responses into 250 or 550 responses that do not contain addresses.

##### 3.6.4.2. VRFY and EXPN Commands and the UTF8REPLY Parameter

If the VRFY and EXPN commands are transmitted with the optional parameter "UTF8REPLY", it indicates the client can accept UTF-8 strings in replies to those commands. This allows the server to use UTF-8 strings in mailbox names and full names that occur in replies without concern that the client might be confused by them. An SMTP client that conforms to this specification MUST accept and correctly process replies from the VRFY and EXPN commands that contain UTF-8 strings. However, the SMTP server MUST NOT use UTF-8 strings in replies if the SMTP client does not specifically allow such replies by transmitting this parameter. Most replies do not require that a mailbox name be included in the returned text, and therefore UTF-8 is not needed in them. Some replies, notably those resulting from successful execution of the VRFY and EXPN commands, do include the mailbox, making the provisions of this section important.

VERIFY (VRFY) and EXPAND (EXPN) command syntaxes are changed to:

```
"VRFY" SP ( uLocal-part / uMailbox ) [ SP "UTF8REPLY" ] CRLF
      ; uLocal-part and uMailbox are defined in
      ; Section 3.3 of this document.
```

```
"EXPN" SP ( uLocal-part / uMailbox ) [ SP "UTF8REPLY" ] CRLF
      ; uLocal-part and uMailbox are defined in
      ; Section 3.3 of this document.
```

The "UTF8REPLY" parameter does not use a value. If the reply to a

VERIFY (VRFY) or EXPAND (EXPN) command requires UTF-8, but the SMTP client did not use the "UTF8REPLY" parameter, then the server MUST use either the response code 252 or 550. Response code 252, defined in [RFC5321], means "Cannot VRFY user, but will accept the message and attempt the delivery". Response code 550, also defined in [RFC5321], means "Requested action not taken: mailbox unavailable". When the server supports enhanced mail system status codes [RFC3463], the enhanced response code as specified below is used. Using the "UTF8REPLY" parameter with a VERIFY (VRFY) or EXPAND (EXPN) command enables UTF-8 replies for that command only.

If a normal success response (i.e., 250) is returned, the response MAY include the full name of the user and MUST include the mailbox of the user. It MUST be in either of the following forms:

```
User Name <uMailbox>
    ; uMailbox is defined in Section 3.3 of this document.
    ; User Name can contain non-ASCII characters.
```

```
uMailbox
    ; uMailbox is defined in Section 3.3 of this document.
```

If the SMTP reply requires UTF-8 strings, but UTF-8 is not allowed in the reply, and the server supports enhanced mail system status codes [RFC3463], the enhanced response code is either "X.6.8" or "X.6.10" [RFC5248], meaning "A reply containing a UTF-8 string is required to show the mailbox name, but that form of response is not permitted by the client".

If the SMTP client does not support the UTF8SMTPbis extension, but receives a UTF-8 string in a reply, it may not be able to properly report the reply to the user, and some clients might crash. Internationalized messages in replies are only allowed in the commands under the situations described above. Under any other circumstances, UTF-8 text MUST NOT appear in the reply.

Although UTF-8 is needed to represent email addresses in responses under the rules specified in this section, this extension does not permit the use of UTF-8 for any other purposes. SMTP servers MUST NOT include non-ASCII characters in replies except in the limited cases specifically permitted in this section.

#### 4. IANA Considerations

IANA should add a new value "UTF8SMTPbis" to the SMTP Service Extension subregistry of the Mail Parameters registry, according to the following data:

Keywords	Description	Reference
UTF8SMTPbis	Internationalized email address	[RFCXXXX]

This document updates the values to the SMTP Enhanced Status Code subregistry of the Mail Parameters registry, following the guidance in Sections 3.4 and 3.6.4.2 of this document, and being based on [RFC5248]. The registration data is as follows:

Code: X.6.7  
Sample Text: UTF-8 addresses not permitted  
for that sender/recipient  
Associated basic status code: 553, 550  
Description: This indicates the reception of a MAIL or RCPT  
command that rUTF-8 addresses are not permitted  
Defined: RFC XXXX (Standard track)  
Submitter: Jiankang YAO  
Change controller: IESG.

Code: X.6.8  
Sample Text: UTF-8 string reply is required,  
but not permitted by the client  
Associated basic status code: 553, 550  
Description: This indicates that a reply containing a UTF-8  
string is required to show the mailbox name,  
but that form of response is not  
permitted by the client.  
Defined: RFC XXXX (Standard track)  
Submitter: Jiankang YAO  
Change controller: IESG.

Code: X.6.9  
Sample Text: UTF-8 header message can not be transferred  
to one or more recipient so the message  
must be bounced  
Associated basic status code: 550  
Description: This indicates that transaction failed  
after the final "." of the DATA command.  
Defined: RFC XXXX (Standard track)  
Submitter: Jiankang YAO  
Change controller: IESG.

Code: X.6.10  
 Sample Text: UTF-8 string reply is required,  
 but not permitted by the client  
 Associated basic status code: 252  
 Description: This indicates that a reply containing a UTF-8  
 string is required to show the mailbox name,  
 but that form of response is not  
 permitted by the client.  
 Defined: RFC XXXX (Standard track)  
 Submitter: Jiankang YAO  
 Change controller: IESG.

The "Mail Transmission Types" registry under the Mail Parameters registry is requested to be updated to include the following new entries:

WITH protocol types	Description	Reference
UTF8SMTPbis	UTF8SMTPbis with Service Extensions	[RFCXXXX]
UTF8SMTPbisA	UTF8SMTPbis with SMTP AUTH	[RFC4954] [RFCXXXX]
UTF8SMTPbisS	UTF8SMTPbis with STARTTLS	[RFC3207] [RFCXXXX]
UTF8SMTPbisSA	UTF8SMTPbis with both STARTTLS and SMTP AUTH	[RFC3207] [RFC4954] [RFCXXXX]

## 5. Security Considerations

See the extended security considerations discussion in the framework document [RFC4952bis].

## 6. Acknowledgements

This document revised the [RFC5336]document based on the EAI WG's discussion result. Many EAI WG members did some tests and implementations to move this document to the Standard Track document. Significant comments and suggestions were received from Xiaodong LEE, Nai-Wen Hsu, Yangwoo KO, Yoshiro YONEYA, and other members of the JET team and were incorporated into the specification. Additional important comments and suggestions, and often specific text, were contributed by many members of the WG and design team. Those contributions include material from John C Klensin, Charles Lindsey, Dave Crocker, Harald Tveit Alvestrand, Marcos Sanz, Chris Newman, Martin Duerst, Edmon Chung, Tony Finch, Kari Hurtta, Randall Gellens,

Frank Ellermann, Alexey Melnikov, Pete Resnick, S. Moonesamy, Soobok Lee, Shawn Steele, Alfred Hoenes, Miguel Garcia, Magnus Westerlund, and Lars Eggert. Of course, none of the individuals are necessarily responsible for the combination of ideas represented here.

## 7. Change History

[[anchor11: RFC Editor: Please remove this section.]]

### 7.1. draft-yao-eai-rfc5336bis: Version 00

Applied errata suggested by Alfred Hoenes.

### 7.2. draft-ietf-eai-rfc5336bis: Version 00

Applied the changes suggested by the EAI new charter.

### 7.3. draft-ietf-eai-rfc5336bis: Version 01

Applied the changes suggested by 78 IETF EAI meeting.

### 7.4. draft-ietf-eai-rfc5336bis: Version 02

remove the appendix since rfc4952bis has added this material

improve the text

remove the text about no body parameter

### 7.5. draft-ietf-eai-rfc5336bis: Version 03

improve the text

### 7.6. draft-ietf-eai-rfc5336bis: Version 04

update the abstract

improve the text

## 8. References

### 8.1. Normative References

[ASCII] American National Standards Institute (formerly United States of America Standards Institute), "USA Code for Information Interchange", ANSI X3.4-1968, 1968.

- [RFC1652] Klensin, J., Freed, N., Rose, M., Stefferud, E., and D. Crocker, "SMTP Service Extension for 8bit-MIMEtransport", RFC 1652, July 1994.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3461] Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, January 2003.
- [RFC3463] Vaudreuil, G., "Enhanced Mail System Status Codes", RFC 3463, January 2003.
- [RFC3464] Moore, K. and G. Vaudreuil, "An Extensible Message Format for Delivery Status Notifications", RFC 3464, January 2003.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", RFC 3629, November 2003.
- [RFC4409] Gellens, R. and J. Klensin, "Message Submission for Mail", RFC 4409, April 2006.
- [RFC4952bis]  
Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", RFC 4952, July 2010.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.
- [RFC5248] Hansen, T. and J. Klensin, "A Registry for SMTP Enhanced Mail System Status Codes", RFC 5248, June 2008.
- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5335bis]  
Abel, Y., Ed., "Internationalized Email Headers", RFC 5335, August 2010.
- [RFC5337bis]  
Newman, C. and A. Melnikov, Ed., "Internationalized Delivery Status and Disposition Notifications", RFC 5337, August 2008.

- [RFC5890] Klensin, J., "Internationalizing Domain Names in Applications (IDNA definitions)", RFC 5890, June 2010.
- [RFC5891] Klensin, J., "Internationalized Domain Names in Applications (IDNA): Protocol", RFC 5891, August 2010.

## 8.2. Informative References

- [RFC0974] Partridge, C., "Mail routing and the domain system", RFC 974, January 1986.
- [RFC2033] Myers, J., "Local Mail Transfer Protocol", RFC 2033, October 1996.
- [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.
- [RFC3030] Vaudreuil, G., "SMTP Service Extensions for Transmission of Large and Binary MIME Messages", RFC 3030, December 2000.
- [RFC3207] Hoffman, P., "SMTP Service Extension for Secure SMTP over Transport Layer Security", RFC 3207, February 2002.
- [RFC4954] Siemborski, R. and A. Melnikov, "SMTP Service Extension for Authentication", RFC 4954, July 2007.
- [RFC5336] Yao, J. and W. Mao, "SMTP Extension for Internationalized Email Addresses", RFC 5336, September 2008.

## Authors' Addresses

Jiankang YAO  
CNNIC  
No.4 South 4th Street, Zhongguancun  
Beijing

Phone: +86 10 58813007  
Email: yaojk@cnnic.cn

Wei MAO  
CNNIC  
No.4 South 4th Street, Zhongguancun  
Beijing

Phone: +86 10 58812230  
Email: maowei\_ietf@cnnic.cn





Network Working Group  
Internet-Draft  
Obsoletes: 5337 (if approved)  
Updates: 3461, 3462, 3464, 3798  
(if approved)  
Intended status: Standards Track  
Expires: April 28, 2011

T. Hansen, Ed.  
AT&T Laboratories  
C. Newman  
Sun Microsystems  
A. Melnikov  
Isode Ltd  
October 25, 2010

Internationalized Delivery Status and Disposition Notifications  
draft-ietf-eai-rfc5337bis-dsn-01

Abstract

Delivery status notifications (DSNs) are critical to the correct operation of an email system. However, the existing Draft Standards (RFC 3461, RFC 3462, RFC 3464) are presently limited to US-ASCII text in the machine-readable portions of the protocol. This specification adds a new address type for international email addresses so an original recipient address with non-US-ASCII characters can be correctly preserved even after downgrading. This also provides updated content return media types for delivery status notifications and message disposition notifications to support use of the new address type.

This document extends RFC 3461, RFC 3462, RFC 3464, and RFC 3798. It replaces the experimental RFC 5337.

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 28, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the

document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Conventions Used in This Document . . . . .	3
3. UTF-8 Address Type . . . . .	3
4. UTF-8 Delivery Status Notifications . . . . .	6
4.1. Additional Requirements on SMTP Servers . . . . .	9
5. UTF-8 Message Disposition Notifications . . . . .	9
6. IANA Considerations . . . . .	11
6.1. UTF-8 Mail Address Type Registration . . . . .	11
6.2. Update to 'smtp' Diagnostic Type Registration . . . . .	12
6.3. message/global-headers . . . . .	12
6.4. message/global-delivery-status . . . . .	13
6.5. message/global-disposition-notification . . . . .	14
7. Security Considerations . . . . .	15
8. References . . . . .	16
8.1. Normative References . . . . .	16
8.2. Informative References . . . . .	17
Appendix A. Changes Since ... . . . .	18
A.1. Changes Since -00 . . . . .	18
A.2. Changes Since RFC 5337 . . . . .	18
Appendix B. Acknowledgements . . . . .	18

## 1. Introduction

When an email message is transmitted using the UTF8SMTP [I-D.ietf-eai-rfc5336bis] extension and Internationalized Email Headers [I-D.ietf-eai-rfc5335bis], it is sometimes necessary to return that message or generate a Message Disposition Notification (MDN) [RFC3798]. As a message sent to multiple recipients can generate a status and disposition notification for each recipient, it is helpful if a client can correlate these notifications based on the recipient address it provided; thus, preservation of the original recipient is important. This specification describes how to preserve the original recipient and updates the MDN and DSN formats to support the new address types.

NOTE: The only issue for which there is (as yet) no consensus yet is whether to change the name of the Address Type from "UTF-8" to something different, such as "UTF8", to reflect the fact that the "<addr <addr>>" address syntax is no longer permitted.

NOTE: There was discussion of whether to change the media type names from message/global, message/global-delivery-status and message/global-headers to something else. The apparent consensus was to not change those names.

## 2. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The formal syntax use the Augmented Backus-Naur Form (ABNF) [RFC5234] notation including the core rules defined in Appendix B of RFC 5234 [RFC5234] and the UTF-8 syntax rules in Section 4 of [RFC3629].

## 3. UTF-8 Address Type

An Extensible Message Format for Delivery Status Notifications [RFC3464] defines the concept of an address type. The address format introduced in Internationalized Email Headers [I-D.ietf-eai-rfc5335bis] is a new address type. The syntax for the new address type in the context of status notifications is specified at the end of this section.

An SMTP [RFC5321] server that advertises both the UTF8SMTP extension [I-D.ietf-eai-rfc5336bis] and the DSN extension [RFC3461] MUST accept a UTF-8 address type in the ORCPT parameter including 8-bit UTF-8 characters. This address type also includes a 7-bit encoding suitable for use in a message/delivery-status body part or an ORCPT

parameter sent to an SMTP server that does not advertise UTF8SMTP.

This address type has 3 forms: utf-8-addr-xttext, utf-8-addr-unitext, and utf-8-address. Only the first form is 7-bit safe.

The utf-8-address form is only suitable for use in newly defined protocols capable of native representation of 8-bit characters. That is, the utf-8-address form MUST NOT be used in the ORCPT parameter when the SMTP server doesn't advertise support for UTF8SMTP, or the SMTP server supports UTF8SMTP, but the address contains US-ASCII characters not permitted in the ORCPT parameter (e.g., the ORCPT parameter forbids unencoded SP and the = character), or in a 7-bit transport environment including a message/delivery-status Original-Recipient or Final-Recipient field. In the first and third case, the utf-8-addr-xttext form (see below) MUST be used instead; in the second case, either the utf-8-addr-unitext or the utf-8-addr-xttext form MUST be used. The utf-8-address form MAY be used in the ORCPT parameter when the SMTP server also advertises support for UTF8SMTP and the address doesn't contain any US-ASCII characters not permitted in the ORCPT parameter. It SHOULD be used in a message/global-delivery-status Original-Recipient or Final-Recipient DSN field, or in an Original-Recipient header field [RFC3798] if the message is a UTF8SMTP message.

In addition, the utf-8-addr-unitext form can be used anywhere where the utf-8-address form is allowed.

When used in the ORCPT parameter, the UTF-8 address type requires that US-ASCII CTLs, SP, \, +, and = be encoded using 'unitext' encoding (see below). This is described by the utf-8-addr-xttext and utf-8-addr-unitext forms in the ABNF below. The 'unitext' encoding uses "\x{HEXPOINT}" syntax (EmbeddedUnicodeChar in the ABNF below) for encoding any Unicode character outside of US-ASCII range, as well as for encoding CTLs, SP, \, +, and =. HEXPOINT is 2 to 6 hexadecimal digits. This encoding avoids the need to use the xttext encoding described in [RFC3461], as any US-ASCII characters that needs to be escaped using xttext encoding never appear in any unitext encoded string. When sending data to a UTF8SMTP capable server, native UTF-8 characters SHOULD be used instead of the EmbeddedUnicodeChar syntax described in details below. When sending data to an SMTP server that does not advertise UTF8SMTP, then the EmbeddedUnicodeChar syntax MUST be used instead of UTF-8.

When the ORCPT parameter is placed in a message/global-delivery-status Original-Recipient field, the 'utf-8-addr-xttext' form of the UTF-8 address type SHOULD be converted to the 'utf-8-address' form (see the ABNF below) by removing the 'unitext' encoding. However, if an address is labeled with the UTF-8 address

type but does not conform to utf-8 syntax, then it MUST be copied into the message/global-delivery-status field without alteration.

The ability to encode characters with the EmbeddedUnicodeChar encodings should be viewed as a transitional mechanism. It is hoped that as systems lacking support for UTF8SMTP become less common over time, these encodings can eventually be phased out.

In the ABNF below, all productions not defined in this document are defined in Appendix B of [RFC5234], in Section 4 of [RFC3629], or in [RFC3464].

```

utf-8-type-addr      = "utf-8;" utf-8-enc-addr

utf-8-address        = uMailbox [ 1*WSP "<" Mailbox ">" ]
    ; uMailbox is defined in [I-D.ietf-eai-rfc5336bis].
    ; Mailbox is defined in [RFC5321].

utf-8-enc-addr       = utf-8-addr-xtext /
    utf-8-addr-unitext /
    utf-8-address

utf-8-addr-xtext     = 1*(QCHAR / EmbeddedUnicodeChar)
    ; 7bit form of utf-8-addr-unitext.
    ; Safe for use in the ORCPT [RFC3461]
    ; parameter even when UTF8SMTP SMTP
    ; extension is not advertised.

utf-8-addr-unitext   = 1*(QUCHAR / EmbeddedUnicodeChar)
    ; MUST follow utf-8-address ABNF when
    ; dequoted.
    ; Safe for using in the ORCPT [RFC3461]
    ; parameter when UTF8SMTP SMTP extension
    ; is also advertised.

QCHAR                = %x21-2a / %x2c-3c / %x3e-5b / %x5d-7e
    ; US-ASCII printable characters except
    ; CTLs, SP, '\', '+', '='.

QUCHAR               = QCHAR / UTF8-2 / UTF8-3 / UTF8-4
    ; US-ASCII printable characters except
    ; CTLs, SP, '\', '+' and '=', plus
    ; other Unicode characters encoded in UTF-8

EmbeddedUnicodeChar  = %x5C.78 "{" HEXPOINT "}"
    ; starts with "\x"

HEXPOINT = ( ( "0"/"1" ) %x31-39 ) / "10" / "20" /

```

```

"2B" / "3D" / "7F" /           ; all xtext-specials
"5C" / (HEXDIG8 HEXDIG) /      ; 2 digit forms
( NZHEXDIG 2(HEXDIG) ) /      ; 3 digit forms
( NZDHEXDIG 3(HEXDIG) ) /      ; 4 digit forms excluding
( "D" %x30-37 2(HEXDIG) ) /   ; ... surrogate
( NZHEXDIG 4(HEXDIG) ) /      ; 5 digit forms
( "10" 4*HEXDIG )             ; 6 digit forms
; represents either "\" or a Unicode code point outside
; the US-ASCII repertoire

HEXDIG8          = %x38-39 / "A" / "B" / "C" / "D" / "E" / "F"
                  ; HEXDIG excluding 0-7
NZHEXDIG         = %x31-39 / "A" / "B" / "C" / "D" / "E" / "F"
                  ; HEXDIG excluding "0"
NZDHEXDIG        = %x31-39 / "A" / "B" / "C" / "E" / "F"
                  ; HEXDIG excluding "0" and "D"

```

#### 4. UTF-8 Delivery Status Notifications

A traditional delivery status notification [RFC3464] comes in a three-part multipart/report [RFC3462] container, where the first part is human-readable text describing the error, the second part is a 7-bit-only message/delivery-status, and the optional third part is used for content (message/rfc822) or header (text/rfc822-headers) return. As the present DSN format does not permit returning of undeliverable UTF8SMTP messages, three new media types are needed.

The first type, message/global-delivery-status, has the syntax of message/delivery-status with three modifications. First, the charset for message/global-delivery-status is UTF-8, and thus any field MAY contain UTF-8 characters when appropriate (see the ABNF below). In particular, the Diagnostic-Code field MAY contain UTF-8 as described in UTF8SMTP [I-D.ietf-eai-rfc5336bis]; the Diagnostic-Code field SHOULD be in i-default language [RFC2277]. Second, systems generating a message/global-delivery-status body part SHOULD use the utf-8-address form of the UTF-8 address type for all addresses containing characters outside the US-ASCII repertoire. These systems SHOULD up-convert the utf-8-addr-xttext or the utf-8-addr-unitext form of a UTF-8 address type in the ORCPT parameter to the utf-8-address form of a UTF-8 address type in the Original-Recipient field. Third, a new optional field called Localized-Diagnostic is added. Each instance includes a language tag [RFC5646] and contains text in the specified language. This is equivalent to the text part of the Diagnostic-Code field. All instances of Localized-Diagnostic MUST use different language tags. The ABNF for message/global-delivery-status is specified below.

In the ABNF below, all productions not defined in this document are

defined in Appendix B of [RFC5234], in Section 4 of [RFC3629], or in [RFC3464]. Note that <text-fixed> is the same as <text> from [RFC5322], but without <obs-text>. If or when RFC 5322 is updated to disallow <obs-text>, this should become just <text> Also, if or when RFC 5322 is updated to disallow control characters in <text>, this should become a reference to that update instead.



```
utf-8-delivery-status-content = per-message-fields
                                1*( CRLF utf-8-per-recipient-fields )
                                ; "per-message-fields" remains unchanged from the definition
                                ; in RFC 3464, except for the "extension-field"
                                ; which is updated below.

utf-8-per-recipient-fields =
    [ original-recipient-field CRLF ]
    final-recipient-field CRLF
    action-field CRLF
    status-field CRLF
    [ remote-mta-field CRLF ]
    [ diagnostic-code-field CRLF
      *(localized-diagnostic-text-field CRLF) ]
    [ last-attempt-date-field CRLF ]
    [ will-retry-until-field CRLF ]
    *( extension-field CRLF )
    ; All fields except for "original-recipient-field",
    ; "final-recipient-field", "diagnostic-code-field"
    ; and "extension-field" remain unchanged from
    ; the definition in RFC 3464.

generic-address =/ utf-8-enc-addr
    ; Only allowed with the "utf-8" address-type.
    ;
    ; This indirectly updates "original-recipient-field"
    ; and "final-recipient-field"

diagnostic-code-field =
    "Diagnostic-Code" ":" diagnostic-type ";" *text-fixed

localized-diagnostic-text-field =
    "Localized-Diagnostic" ":" Language-Tag ";" *utf8-text
    ; "Language-Tag" is a language tag as defined in [LANGTAGS].

extension-field =/ extension-field-name ":" *utf8-text

text-fixed = %d1-9 /           ; Any US-ASCII character except for NUL,
                        %d11 /   ; CR and LF
                        %d12 /   ; See note above about <text-fixed>
                        %d14-127

utf8-text = text-fixed / UTF8-non-ascii

UTF8-non-ascii = UTF8-2 / UTF8-3 / UTF8-4

The second type, used for returning the content, is message/global
which is similar to message/rfc822, except it contains a message with
```

UTF-8 headers. This media type is described in [I-D.ietf-eai-rfc5335bis].

The third type, used for returning the headers, is message/global-headers and contains only the UTF-8 header fields of a message (all lines prior to the first blank line in a UTF8SMTP message). Unlike message/global, this body part provides no difficulties for the present infrastructure.

Note that as far as multipart/report [RFC3462] container is concerned, message/global-delivery-status, message/global, and message/global-headers MUST be treated as equivalent to message/delivery-status, message/rfc822, and text/rfc822-headers. That is, implementations processing multipart/report MUST expect any combinations of the 6 media types mentioned above inside a multipart/report media type.

All three new types will typically use the "8bit" Content-Transfer-Encoding. (In the event all content is 7-bit, the equivalent traditional types for delivery status notifications MAY be used. For example, if information in message/global-delivery-status part can be represented without any loss of information as message/delivery-status, then the message/delivery-status body part may be used.) Note that [I-D.ietf-eai-rfc5335bis] relaxed restriction from MIME [RFC2046] regarding use of Content-Transfer-Encoding in new "message" subtypes. This specification explicitly allows use of Content-Transfer-Encoding in message/global-headers and message/global-delivery-status. This is not believed to be problematic as these new media types are intended primarily for use by newer systems with full support for 8-bit MIME and UTF-8 headers.

#### 4.1. Additional Requirements on SMTP Servers

If an SMTP server that advertises both UTF8SMTP and DSN needs to return an undeliverable UTF8SMTP message, then it MUST NOT downgrade [RFC5504] the UTF8SMTP message when generating the corresponding multipart/report. If the return path SMTP server does not support UTF8SMTP, then the undeliverable body part and headers MUST be encoded using a 7-bit Content-Transfer-Encoding such as "base64" or "quoted-printable" [RFC2045], as detailed in Section 4. Otherwise, "8bit" Content-Transfer-Encoding can be used.

#### 5. UTF-8 Message Disposition Notifications

Message Disposition Notifications [RFC3798] have a similar design and structure to DSNs. As a result, they use the same basic return format. When generating an MDN for a UTF-8 header message, the third part of the multipart/report contains the returned content (message/

global) or header (message/global-headers), same as for DSNs. The second part of the multipart/report uses a new media type, message/global-disposition-notification, which has the syntax of message/disposition-notification with two modifications. First, the charset for message/global-disposition-notification is UTF-8, and thus any field MAY contain UTF-8 characters when appropriate (see the ABNF below). (In particular, the failure-field, the error-field, and the warning-field MAY contain UTF-8. These fields SHOULD be in i-default language [RFC2277].) Second, systems generating a message/global-disposition-notification body part (typically a mail user agent) SHOULD use the UTF-8 address type for all addresses containing characters outside the US-ASCII repertoire.

The MDN specification also defines the Original-Recipient header field, which is added with a copy of the contents of ORCPT at delivery time. When generating an Original-Recipient header field, a delivery agent writing a UTF-8 header message in native format SHOULD convert the utf-8-addr-xttext or the utf-8-addr-unitext form of a UTF-8 address type in the ORCPT parameter to the corresponding utf-8-address form.

The MDN specification also defines the Disposition-Notification-To header field, which is an address header field and thus follows the same 8-bit rules as other address header fields such as "From" and "To" when used in a UTF-8 header message.

```
; ABNF for "original-recipient-header", "original-recipient-field",  
; and "final-recipient-field" from RFC 3798 is implicitly updated  
; as they use the updated "generic-address" as defined in  
; Section 4 of this document.
```

```
failure-field = "Failure" ":" *utf8-text  
; "utf8-text" is defined in Section 4 of this document.
```

```
error-field = "Error" ":" *utf8-text  
; "utf8-text" is defined in Section 4 of this document.
```

```
warning-field = "Warning" ":" *utf8-text  
; "utf8-text" is defined in Section 4 of this document.
```

## 6. IANA Considerations

This specification does not create any new IANA registries. However, the following items are registered as a result of this document.

### 6.1. UTF-8 Mail Address Type Registration

The mail address type registry was created by [RFC3464]. The registration template response follows:

- (a) The proposed address-type name.

UTF-8

- (b) The syntax for mailbox addresses of this type, specified using BNF, regular expressions, ASN.1, or other non-ambiguous language.

See Section 3.

- (c) If addresses of this type are not composed entirely of graphic characters from the US-ASCII repertoire, a specification for how they are to be encoded as graphic US-ASCII characters in a DSN Original-Recipient or Final-Recipient DSN field.

This address type has 3 forms (as defined in Section 3): utf-8-addr-xtext, utf-8-addr-unitext, and utf-8-address. Only the first form is 7-bit safe.

The utf-8-address form MUST NOT be used

1. in the ORCPT parameter when the SMTP server doesn't advertise support for UTF8SMTP;
2. or the SMTP server supports UTF8SMTP, but the address contains US-ASCII characters not permitted in the ORCPT parameter (e.g., the ORCPT parameter forbids SP and the = characters);
3. or in a 7-bit transport environment including a message/delivery-status Original-Recipient or Final-Recipient field.

The utf-8-addr-xtext form MUST be used instead in the first and the third case; the utf-8-addr-unitext form MUST be used in the second case. The utf-8-address form MAY be used in the ORCPT parameter when the SMTP server also advertises support for UTF8SMTP and the address doesn't contain any US-ASCII characters not permitted in the ORCPT parameter; in a message/global-delivery-status Original-Recipient or Final-Recipient DSN field; or in an Original-Recipient header field [RFC3798] if the message is a UTF8SMTP message.

In addition, the utf-8-addr-unitext form can be used anywhere where the utf-8-address form is allowed.

## 6.2. Update to 'smtp' Diagnostic Type Registration

The mail diagnostic type registry was created by [RFC3464] and updated by [RFC5337]. The registration for the 'smtp' diagnostic type should be updated to reference RFC XXXX in addition to [RFC3464] and [RFC5337].

When the 'smtp' diagnostic type is used in the context of a message/delivery-status body part, it remains as presently defined. When the 'smtp' diagnostic type is used in the context of a message/global-delivery-status body part, the codes remain the same, but the text portion MAY contain UTF-8 characters.

## 6.3. message/global-headers

Type name: message

Subtype name: global-headers

Required parameters: none

Optional parameters: none

Encoding considerations: This media type contains Internationalized Email Headers [I-D.ietf-eai-rfc5335bis] with no message body. Whenever possible, the 8-bit content transfer encoding SHOULD be used. When this media type passes through a 7-bit-only SMTP infrastructure it MAY be encoded with the base64 or quoted-printable content transfer encoding.

Security considerations: See Section 7.

Interoperability considerations: It is important that this media type is not converted to a charset other than UTF-8. As a result, implementations MUST NOT include a charset parameter with this media type. Although it might be possible to downconvert this media type to the text/rfc822-header media type, such conversion is discouraged as it loses information.

Published specification: RFC XXXX

Applications that use this media type: UTF8SMTP servers and email clients that support multipart/report generation or parsing.

Additional information:

Magic number(s): none

File extension(s): In the event this is saved to a file, the extension ".u8hdr" is suggested.

Macintosh file type code(s): The 'TEXT' type code is suggested as files of this type are typically used for diagnostic purposes and suitable for analysis in a UTF-8 aware text editor. A uniform type identifier (UTI) of "public.utf8-email-message-header" is suggested. This type conforms to "public.utf8-plain-text" and "public.plain-text".

Person & email address to contact for further information: See the Authors' Addresses section of this document.

Intended usage: COMMON

Restrictions on usage: This media type contains textual data in the UTF-8 charset. It typically contains octets with the 8th bit set. As a result, a transfer encoding is required when a 7-bit transport is used.

Author: See the Authors' Addresses section of this document.

Change controller: IETF Standards Process

#### 6.4. message/global-delivery-status

Type name: message

Subtype name: global-delivery-status

Required parameters: none

Optional parameters: none

Encoding considerations: This media type contains delivery status notification attributes in the UTF-8 charset. The 8-bit content transfer encoding MUST be used with this content-type, unless it is sent over a 7-bit transport environment in which case quoted-printable or base64 may be necessary.

Security considerations: See Section 7

Interoperability considerations: This media type provides functionality similar to the message/delivery-status content-type for email message return information. Clients of the previous format will need to be upgraded to interpret the new format; however, the new media type makes it simple to identify the difference.

Published specification: RFC XXXX

Applications that use this media type: SMTP servers and email clients that support delivery status notification generation or parsing.

Additional information:

Magic number(s): none

File extension(s): The extension ".u8dsn" is suggested.

Macintosh file type code(s): A uniform type identifier (UTI) of "public.utf8-email-message-delivery-status" is suggested. This type conforms to "public.utf8-plain-text".

Person & email address to contact for further information: See the Authors' Addresses section of this document.

Intended usage: COMMON

Restrictions on usage: This is expected to be the second part of a multipart/report.

Author: See the Authors' Addresses section of this document.

Change controller: IETF Standards Process

#### 6.5. message/global-disposition-notification

Type name: message

Subtype name: global-disposition-notification

Required parameters: none

Optional parameters: none

Encoding considerations: This media type contains disposition notification attributes in the UTF-8 charset. The 8-bit content transfer encoding **MUST** be used with this content-type, unless it is sent over a 7-bit transport environment in which case quoted-printable or base64 may be necessary.

Security considerations: See Section 7.

Interoperability considerations: This media type provides functionality similar to the message/disposition-notification content-type for email message disposition information. Clients of the previous format will need to be upgraded to interpret the new format; however, the new media type makes it simple to identify the difference.

Published specification: RFC XXXX

Applications that use this media type: Email clients or servers that support message disposition notification generation or parsing.

Additional information:

Magic number(s): none

File extension(s): The extension ".u8mdn" is suggested.

Macintosh file type code(s): A uniform type identifier (UTI) of "public.utf8-email-message-disposition-notification" is suggested. This type conforms to "public.utf8-plain-text".

Person & email address to contact for further information: See the Authors' Addresses section of this document.

Intended usage: COMMON

Restrictions on usage: This is expected to be the second part of a multipart/report.

Author: See the Authors' Addresses section of this document.

Change controller: IETF Standards Process

## 7. Security Considerations

Automated use of report types without authentication presents several security issues. Forging negative reports presents the opportunity for denial-of-service attacks when the reports are used for automated maintenance of directories or mailing lists. Forging positive



reports may cause the sender to incorrectly believe a message was delivered when it was not.

Malicious users can generate report structures designed to trigger coding flaws in report parsers. Report parsers need to use secure coding techniques to avoid the risk of buffer overflow or denial-of-service attacks against parser coding mistakes. Code reviews of such parsers are also recommended.

Malicious users of the email system regularly send messages with forged envelope return paths, and these messages trigger delivery status reports that result in a large amount of unwanted traffic on the Internet. Many users choose to ignore delivery status notifications because they are usually the result of "blowback" from forged messages and thus never notice when messages they sent go undelivered. As a result, support for correlation of delivery status and message disposition notification messages with sent-messages has become a critical feature of mail clients and possibly mail stores if the email infrastructure is to remain reliable. In the short term, simply correlating message-IDs may be sufficient to distinguish true status notifications from those resulting from forged originator addresses. But in the longer term, including cryptographic signature material that can securely associate the status notification with the original message is advisable.

As this specification permits UTF-8 in additional fields, the security considerations of UTF-8 [RFC3629] apply.

## 8. References

### 8.1. Normative References

- |           |   |
|-----------|---|
| [RFC2119] | Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.                                |
| [RFC2277] | Alvestrand, H., "IETF Policy on Character Sets and Languages", BCP 18, RFC 2277, January 1998.  |
| [RFC3461] | Moore, K., "Simple Mail Transfer Protocol (SMTP) Service Extension for Delivery Status Notifications (DSNs)", RFC 3461, January 2003. |
| [RFC3462] | Vaudreuil, G., "The Multipart/Report Content Type for the Reporting of Mail System Administrative Messages", RFC 3462,                |

January 2003.

- [RFC3464] Moore, K. and G. Vaudreuil, "An Extensible Message Format for Delivery Status Notifications", RFC 3464, January 2003.
- [RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.
- [RFC3798] Hansen, T. and G. Vaudreuil, "Message Disposition Notification", RFC 3798, May 2004.
- [RFC5646] Phillips, A. and M. Davis, "Tags for Identifying Languages", BCP 47, RFC 5646, September 2009.
- [RFC5321] Klensin, J., "Simple Mail Transfer Protocol", RFC 5321, October 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.
- [I-D.ietf-eai-rfc5335bis] Yang, A. and S. Steele, "Internationalized Email Headers", draft-ietf-eai-rfc5335bis-03 (work in progress), October 2010.
- [I-D.ietf-eai-rfc5336bis] Yao, J. and W. MAO, "SMTP Extension for Internationalized Email Address", draft-ietf-eai-rfc5336bis-04 (work in progress), October 2010.

## 8.2. Informative References

- [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.

- [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.

## Appendix A. Changes Since ...

### A.1. Changes Since -00

Incorporated changes from draft-ietf-eai-dsnbis-01.

Fixed description of utf-8-addr-xtxt and utf-8-addr-unitext.

Other minor corrections.

### A.2. Changes Since RFC 5337

Made minor changes to move from Experimental to Standards Track.

Minor ABNF changes and errata suggested by Alfred Hoenes.

Minor changes to MIME type references.

Other minor corrections.

## Appendix B. Acknowledgements

Many thanks for input provided by Pete Resnick, James Galvin, Ned Freed, John Klensin, Harald Alvestrand, Frank Ellermann, SM, Alfred Hoenes, Kazunori Fujiwara, and members of the EAI WG to help solidify this proposal.

## Authors' Addresses

Tony Hansen (editor)  
AT&T Laboratories  
200 Laurel Ave.  
Middletown, NJ 07748  
USA

Email: [tony+eaidns@maillennium.att.com](mailto:tony+eaidns@maillennium.att.com)

Chris Newman  
Sun Microsystems  
800 Royal Oaks  
Monrovia, CA 91016-6347  
US

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

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

EMail: [Alexey.Melnikov@isode.com](mailto:Alexey.Melnikov@isode.com)



Network Working Group  
Internet-Draft  
Obsoletes: RFC5721  
(if approved)  
Updates: RFC1939  
(if approved)  
Intended status: Standards Track  
Expires: March 14, 2011

R. Gellens  
QUALCOMM Incorporated  
C. Newman  
Oracle  
Jiankang. Yao  
CNNIC  
Kazunori. Fujiwara  
JPRS  
September 28, 2010

POP3 Support for UTF-8  
draft-ietf-eai-rfc5721bis-00.txt

Abstract

This specification extends the Post Office Protocol version 3 (POP3) to support un-encoded international characters in user names, passwords, mail addresses, message headers, and protocol-level textual error strings.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 14, 2011.

Copyright Notice

Copyright (c) 2010 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect

to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF 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
1.1. Conventions Used in This Document . . . . .	3
2. LANG Capability . . . . .	4
3. UTF8 Capability . . . . .	6
3.1. The UTF8 Command . . . . .	7
3.2. USER Argument to UTF8 Capability . . . . .	8
4. Native UTF-8 Maildrops . . . . .	9
5. IANA Considerations . . . . .	9
6. Security Considerations . . . . .	9
7. References . . . . .	10
7.1. Normative References . . . . .	10
7.2. Informative References . . . . .	11
Appendix A. Design Rationale . . . . .	12
Appendix B. Acknowledgments . . . . .	12

## 1. Introduction

This document forms part of the Email Address Internationalization (EAI) protocols described in the EAI Framework document [I-D.ietf-eai-frmwrk-4952bis]. As part of the overall EAI work, email messages may be transmitted and delivered containing un-encoded UTF-8 characters, and mail drops that are accessed using POP3 [RFC1939] might natively store UTF-8.

This specification extends POP3 [RFC1939] using the POP3 extension mechanism [RFC2449] to permit un-encoded UTF-8 [RFC3629] in headers, as described in "Internationalized Email Headers" [I-D.ietf-eai-rfc5335bis]. It also adds a mechanism to support login names and passwords outside the ASCII character set, and a mechanism to support UTF-8 protocol-level error strings in a language appropriate for the user.

Within this specification, the term "down-conversion" refers to the process of modifying a message containing UTF-8 headers [I-D.ietf-eai-rfc5335bis] or body parts with 8bit content-transfer-encoding, as defined in MIME Section 2.8 [RFC2045], into conforming 7-bit Internet Message Format [RFC5322] with message header extensions for non-ASCII text [RFC2047] and other 7-bit encodings. Down-conversion is specified by "Message-Downgrading for Email Address Internationalization" [message-downgrade].

### 1.1. Conventions Used in This Document

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 "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 RFC 5234.

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.

Note that examples always use 7-bit ASCII characters due to limitations of this document format; in particular, some examples for the "LANG" command may appear silly as a result.



## 2. LANG Capability

Per "POP3 Extension Mechanism" [RFC2449], this document adds a new capability response tag to indicate support for a new command: LANG. The capability tag and new command are described below.

CAPA tag:  
LANG

Arguments with CAPA tag:  
none

Added Commands:  
LANG

Standard commands affected:  
All

Announced states / possible differences:  
both / no

Commands valid in states:  
AUTHENTICATION, TRANSACTION

Specification reference:  
this document

Discussion:

POP3 allows most +OK and -ERR server responses to include human-readable text that, in some cases, might be presented to the user. But that text is limited to ASCII by the POP3 specification [RFC1939]. The LANG capability and command permit a POP3 client to negotiate which language the server should use when sending human-readable text.

A server that advertises the LANG extension MUST use the language "i-default" as described in [RFC2277] as its default language until another supported language is negotiated by the client. A server MUST include "i-default" as one of its supported languages.

The LANG command requests that human-readable text included in all subsequent +OK and -ERR responses be localized to a language matching the language range argument (the "Basic Language Range" as described by [RFC4647]). If the command succeeds, the server returns a +OK response followed by a single space, the exact language tag selected, another space, and the rest of the line is human-readable text in the appropriate language. This and subsequent protocol-level human-

readable text is encoded in the UTF-8 charset.

If the command fails, the server returns an -ERR response and subsequent human-readable response text continues to use the language that was previously active (typically i-default).

The special "\*" language range argument indicates a request to use a language designated as preferred by the server administrator. The preferred language MAY vary based on the currently active user.

If no argument is given and the POP3 server issues a positive response, then the response given is multi-line. After the initial +OK, for each language tag the server supports, the POP3 server responds with a line for that language. This line is called a "language listing".

In order to simplify parsing, all POP3 servers are required to use a certain format for language listings. A language listing consists of the language tag [RFC5646] of the message, optionally followed by a single space and a human-readable description of the language in the language itself, using the UTF-8 charset.

Examples:

< Note that some examples do not include the correct character accents due to limitations of this document format. >

< The server defaults to using English i-default responses until the client explicitly changes the language. >

```
C: USER karen
S: +OK Hello, karen
C: PASS password
S: +OK karen's maildrop contains 2 messages (320 octets)
```

< Client requests deprecated MUL language. Server replies with -ERR response. >

```
C: LANG MUL
S: -ERR invalid language MUL
```

< A LANG command with no parameters is a request for a language listing. >

```
C: LANG
S: +OK Language listing follows:
S: en English
S: en-boont English Boontling dialect
```

```
S: de Deutsch
S: it Italiano
S: es Espanol
S: sv Svenska
S: i-default Default language
S: .
```

< A request for a language listing might fail. >

```
C: LANG
S: -ERR Server is unable to list languages
```

< Once the client changes the language, all responses will be in that language, starting with the response to the LANG command. >

```
C: LANG es
S: +OK es Idioma cambiado
```

< If a server does not support the requested primary language, responses will continue to be returned in the current language the server is using. >

```
C: LANG uga
S: -ERR es Idioma <<UGA>> no es conocido
```

```
C: LANG sv
S: +OK sv Kommandot "LANG" lyckades
```

```
C: LANG *
S: +OK es Idioma cambiado
```

### 3. UTF8 Capability

Per "POP3 Extension Mechanism" [RFC2449], this document adds a new capability response tag to indicate support for new server functionality, including a new command: UTF8. The capability tag and new command and functionality are described below.

CAPA tag:  
UTF8

Arguments with CAPA tag:  
USER

Added Commands:  
UTF8

Standard commands affected:

USER, PASS, APOP, LIST, TOP, RETR

Announced states / possible differences:

both / no

Commands valid in states:

AUTHORIZATION

Specification reference:

this document

Discussion:

This capability adds the "UTF8" command to POP3. The UTF8 command switches the session from ASCII to UTF-8 mode.

### 3.1. The UTF8 Command

The UTF8 command enables UTF-8 mode. The UTF8 command has no parameters.

Maildrops can natively store UTF-8 or be limited to ASCII. UTF-8 mode has no effect on messages in an ASCII-only maildrop. Messages in native UTF-8 maildrops can be ASCII or UTF-8 using internationalized headers [I-D.ietf-eai-rfc5335bis] and/or 8bit content-transfer-encoding, as defined in MIME Section 2.8 [RFC2045]. In UTF-8 mode, both UTF-8 and ASCII messages are sent to the client as-is (without conversion). When not in UTF-8 mode, UTF-8 messages in a native UTF-8 maildrop MUST NOT be sent to the client as-is. UTF-8 messages in a native UTF-8 maildrop MUST be down-converted (downgraded) to comply with unextended POP and Internet Mail Format without UTF-8 mode support.

Note that even in UTF-8 mode, MIME binary content-transfer-encoding is still not permitted.

The octet count (size) of a message reported in a response to the LIST command SHOULD match the actual number of octets sent in a RETR response (not counting byte-stuffing). Sizes reported elsewhere, such as in STAT responses and non-standardized, free-form text in positive status indicators (following "+OK") need not be accurate, but it is preferable if they are.

Mail stores are either ASCII or native UTF-8, and clients either issue the UTF8 command or not. The message needs converting only when it is native UTF-8 and the client has not issued the UTF8 command, in which case the server must down-convert it. The down-

converted message may be larger. The server may choose various strategies regarding down-conversion, which include when to down-convert, whether to cache or store the down-converted form of a message (and if so, for how long), and whether to calculate or retain the size of a down-converted message independently of the down-converted content. If the server does not have immediate access to the accurate down-converted size, it may be faster to estimate rather than calculate it. Servers are expected to normally follow the RFC 1939 [RFC1939] text on using the "exact size" in a scan listing, but there may be situations with maildrops containing very large numbers of messages in which this might be a problem. If the server does estimate, reporting a scan listing size smaller than what it turns out to be could be a problem for some clients. In summary, it is better for servers to report accurate sizes, but if this is not possible, high guesses are better than small ones. Some POP servers include the message size in the non-standardized text response following "+OK" (the 'text' production of RFC 2449 [RFC2449]), in a RETR or TOP response (possibly because some examples in POP3 [RFC1939] do so). There has been at least one known case of a client relying on this to know when it had received all of the message rather than following the POP3 [RFC1939] rule of looking for a line consisting of a termination octet (".") and a CRLF pair. While any such client is non-compliant, if a server does include the size in such text, it is better if it is accurate.

Clients MUST NOT issue the STLS command [RFC2595] after issuing UTF8; servers MAY (but are not required to) enforce this by rejecting with an "-ERR" response an STLS command issued subsequent to a successful UTF8 command. (Because this is a protocol error as opposed to a failure based on conditions, an extended response code [RFC2449] is not specified.)

### 3.2. USER Argument to UTF8 Capability

If the USER argument is included with this capability, it indicates that the server accepts UTF-8 user names and passwords.

Servers that include the USER argument in the UTF8 capability response SHOULD apply SASLprep [RFC4013] to the arguments of the USER and PASS commands.

A client or server that supports APOP and permits UTF-8 in user names or passwords MUST apply SASLprep [RFC4013] to the user name and password used to compute the APOP digest.

When applying SASLprep [RFC4013], servers MUST reject UTF-8 user names or passwords that contain a Unicode character listed in Section 2.3 of SASLprep [RFC4013]. When applying SASLprep to the USER

argument, the PASS argument, or the APOP username argument, a compliant server or client MUST treat them as a query string (i.e., unassigned Unicode code points are allowed). When applying SASLprep to the APOP password argument, a compliant server or client MUST treat them as a stored string (i.e., unassigned Unicode code points are prohibited).

The client does not need to issue the UTF8 command prior to using UTF-8 in authentication. However, clients MUST NOT use UTF-8 characters in USER, PASS, or APOP commands unless the USER argument is included in the UTF8 capability response.

The server MUST reject UTF-8 user names or passwords that fail to comply with the formal syntax in UTF-8 [RFC3629].

Use of UTF-8 characters in the AUTH command is governed by the POP3 SASL [RFC5034] mechanism.

#### 4. Native UTF-8 Maildrops

When a POP3 server uses a native UTF-8 maildrop, it is the responsibility of the server to comply with the POP3 base specification [RFC1939] and Internet Message Format [RFC5322] when not in UTF-8 mode. Mechanisms for 7-bit downgrading to help comply with the standards are described in [message-downgrade].

#### 5. IANA Considerations

This specification adds two new capabilities ("UTF8" and "LANG") to the POP3 capability registry [RFC2449].

#### 6. 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.

The "LANG \*" command might reveal the existence and preferred language of a user to an active attacker probing the system if the active language changes in response to the USER, PASS, or APOP commands prior to validating the user's credentials. Servers MUST implement a configuration to prevent this exposure.

It is possible for a man-in-the-middle attacker to insert a LANG command in the command stream, thus making protocol-level diagnostic responses unintelligible to the user. A mechanism to integrity-protect the session, such as Transport Layer Security (TLS) [RFC2595] can be used to defeat such attacks.

Modifying server authentication code (in this case, to support UTF8 command) needs to be done with care to avoid introducing vulnerabilities (for example, in string parsing).

The UTF8 command description (Section 3.1) contains a discussion on reporting inaccurate sizes. An additional risk to doing so is that, if a client allocates buffers based on the reported size, it may overrun the buffer, crash, or have other problems if the message data is larger than reported.

## 7. References

### 7.1. Normative References

- [I-D.ietf-eai-frmwrk-4952bis] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", draft-ietf-eai-frmwrk-4952bis-07 (work in progress), August 2010.
- [I-D.ietf-eai-rfc5335bis] Yang, A. and S. Steele, "Internationalized Email Headers", draft-ietf-eai-rfc5335bis-02 (work in progress), August 2010.
- [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.
- [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.
- [RFC2277] Alvestrand, H., "IETF Policy on Character Sets and Languages", BCP 18, RFC 2277, January 1998.

- [RFC2449] Gellens, R., Newman, C., and L. Lundblade, "POP3 Extension Mechanism", RFC 2449, November 1998.
- [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.
- [RFC4647] Phillips, A. and M. Davis, "Matching of Language Tags", BCP 47, RFC 4647, September 2006.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.
- [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, October 2008.
- [RFC5646] Phillips, A. and M. Davis, "Tags for Identifying Languages", BCP 47, RFC 5646, September 2009.

## 7.2. Informative References

- [RFC2595] Newman, C., "Using TLS with IMAP, POP3 and ACAP", RFC 2595, June 1999.
- [RFC4952] Klensin, J. and Y. Ko, "Overview and Framework for Internationalized Email", RFC 4952, July 2007.
- [RFC5034] Siemborski, R. and A. Menon-Sen, "The Post Office Protocol (POP3) Simple Authentication and Security Layer (SASL) Authentication Mechanism", RFC 5034, July 2007.
- [message-downgrade] Fujiwara, K. and Y. Yoneya, "Message Downgrading for Email Address Internationalization (EAI) Maildrops", draft-ietf-eai-rfc5504bis-00 (work in progress), Sep 2010.



## Appendix A. Design Rationale

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

Due to interoperability problems with RFC 2047 and limited deployment of RFC 2231, it is hoped these 7-bit encoding mechanisms can be deprecated in the future when UTF-8 header support becomes prevalent.

USER is optional because the implementation burden of SASLprep [RFC4013] is not well understood, and mandating such support in all cases could negatively impact deployment.

While it is possible to provide useful examples for language negotiation without support for non-ASCII characters, it is difficult to provide useful examples for commands specifically designed to use the UTF-8 charset un-encoded when the document format is limited to ASCII. As a result, there are no plans to provide examples for that part of the specification as long as this remains an experimental proposal. However, implementers of this specification are encouraged to provide examples to the document authors for a future revision.

## Appendix B. Acknowledgments

Thanks to John Klensin, Tony Hansen, and other EAI working group participants who provided helpful suggestions and interesting debate that improved this specification.

## Authors' Addresses

Randall Gellens  
QUALCOMM Incorporated  
5775 Morehouse Drive  
San Diego, CA 92651  
US  
  
EMail: rg+ietf@qualcomm.com

Chris Newman  
Oracle  
800 Royal Oaks  
Monrovia, CA 91016-6347  
US  
  
EMail: chris.newman@oracle.com

Jiankang YAO  
CNNIC  
No.4 South 4th Street, Zhongguancun  
Beijing

Phone: +86 10 58813007  
EMail: yaojk@cnnic.cn

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

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

