Network Working Group
Internet-Draft
Expires: January 6, 2008

Y. Abel, Ed.
TWNIC

July 05, 2007

Internationalized Email Headers
draft-ietf-eai-utf8headers-06.txt

Status of this Memo

Copyright Notice

Abstract

Full internationalization of electronic mail requires not only the capability 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 information based on them in mail header fields. This document specifies an experimental 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 bodies. This form is permitted in transmission only if authorized by an SMTP

extension, as specified in an associated specification.

Table of Contents

1. Introduction

1.1. Role of this specification

Full internationalization of electronic mail requires several capabilities:

- The capability to transmit non-ASCII content, provided for as part of the basic MIME specification [RFC2045], [RFC2046].
- The capability to express those addresses, and information related to and based on them, in mail header fields, defined in this document. And, finally,
- The capability to use international characters in envelope addresses, discussed in [EAI-framework] and specified in [EAI-SMTP-extension].
-

This document specifies an experimental 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 [EAI-SMTP-extension] or by other transport mechanisms capable of processing it.

1.2. Relation to other standards

This document updates section 6.4 of RFC 2045. 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 update [RFC2822] and MIME, and the fact that an experimental specification updates a standards-track spec means that people who participate in the experiment have to consider those standards updated.

Allowing of use a content-transfer-encoding on subtypes of messages is not limited to transmissions, which are authorized by the SMTP extension specified in [EAI-SMTP-extension]. Message/utf8smtp permits use of a content-transfer-encoding. ([AUTHOR NOTE] The message/utf8smtp is a placeholder for whatever the WG decides to call it)

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 in what they send and what they receive. This protocol specifies UTF-8 as the encoding to represent email header field bodies.

The traditional format of email messages [RFC2822] 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 document [EAI-framework] and [EAI-SMTP-extension], 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 [EAI-SMTP-extension], an SMTP protocol extension "UTF8SMTP" is used to prevent the transmission of messages with UTF-8 header fields to systems that cannot handle such messages.

Use of this SMTP extension helps prevents 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 transfering 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 related documents.

The objective for this protocol is to allow UTF-8 in email header fields. Issues about how to handle messages that contain UTF-8 header fields but are proposed to be delivered to systems that have not been upgraded to support this capability are discussed elsewhere, particularly in [EAI-downgrading].

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 [RFC2821]

or [RFC2822] or in [EAI-framework].

The key words "MUST", "SHALL", "REQUIRED", "SHOULD", "RECOMMENDED", and "MAY" in this document are to be interpreted as described in [RFC2119].

This document is discussed on the ima mailing list. See https://www1.ietf.org/mailman/listinfo/ima for information about subscribing. The list's archive is at http://www1.ietf.org/mail-archive/web/ima/index.html.

4. Changes on Message Header Fields

SMTP clients can send header fields in UTF-8 format, if the UTF8SMTP extension is advertised by the SMTP server or as permitted by other transport mechanisms.

This protocol does NOT change the definition of header field names. That is, only the bodies of header fields are allowed to have UTF-8 characters; the rules in [RFC2822] for header field names are not changed.

To permit UTF-8 characters in field values, the header definition in [RFC2822] must be extended to support new format. The following ABNF is defined to substitute those definition in [RFC2822].

Those syntax rules not referred to this section remain as the original definition in [RFC2822].

4.1. UTF8 Syntax

UTF-8 characters can be defined in terms of octets using the following ABNF, 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-F7 3(UTF8-tail)

UTF8-tail       =   %x80-BF
```

These are taken from [RFC3629], but kept in this document for reasons of convenience.
[Note in draft: Whether normalizing is needed or not will be place in here.]

4.2. Changes on MIME headers

This specification updates section 6.4 of RFC 2045. RFC 2045 prohibits applying a content-transfer-encoding to all subtypes of message/. This specification relaxes the rule, permitting content-transfer-encoding for message/utf8smtp only (see Section 4.6). Normally, transfer of message/utf8smtp will be done in 8-bit-clean channels, and body parts will have 8-bit encodings. The additional complexity of a content-transfer-encoding on "message" is therefore acceptable. This specification does not prohibit other content-transfer-encodings on nested body parts, so double encoding might happen, but is expected to be rarely seen in practice.

To be able to use UTF-8 characters in MIME header field paramerer values, the syntax of <value> , as defined in [RFC2045], is extended as

```
value   =/      utf8-quoted-string
```

Because of MIME's structure, Content-Type and other header fields may be found both amongst the top-level fields of a message and also within its body parts.

4.3. Syntax extensions to RFC 2822

The following rules are intended to extend the corresponding rules in [RFC2822] to allow UTF8 characters.

```
ctext   =/  UTF8-xtra-char

utext   =/  UTF8-xtra-char

comment = "(" *([FWS] utf8-ccontent) [FWS] ")"

word    = utf8-atom / utf8-quoted-string
```

This means that all the [RFC2822] constructs that build upon these will permit UTF-8 characters, including comments and quoted strings. Besides, in order to allow UTF8 characters in <addr-spec> we have to change the syntax of <atext>. However, it would also lead <msg-id> to allow UTF8 characters, which is not allowed due to the limitation

described in Section 4.5. So <utf8-atext> is added to meet this requirement.

```
utf8-text   =  %d1-9 /         ; all UTF-8 characters except
           %d11-12 /       ; US-ASCII NUL, CR and LF
           %d14-127 /
           UTF8-xtra-char

utf8-quoted-pair   = ("\" utf8-text) / obs-qp

utf8-qcontent      = utf8-qtext / utf8-quoted-pair

utf8-quoted-string = [CFWS]
                        DQUOTE *([FWS] utf8-qcontent) [FWS] DQUOTE
                        [CFWS]

utf8-ccontent = ctext / utf8-quoted-pair / comment
utf8-qtext= qtext / UTF8-xtra-char
qtext     =     NO-WS-CTL /     ; all of <text> except
           %d33 /              ; The rest of the US-ASCII
           %d35-91 /        ; characters not including "\"
           %d93-126 /       ; or the quote character


utf8-atext   =  ALPHA / DIGIT /
                "!" / "#" /     ; Any character except
                "$" / "%" /     ; controls, SP, and specials.
                "&" / "'" /     ; Used for atoms
                "*" / "+" /
                "-" / "/" /
                "=" / "?" /
                "^" / "_" /
                "`" / "{" /
                "|" / "}" /
                "~" /
                UTF8-xtra-char

utf8-atom     = [CFWS] 1*utf8-atext [CFWS]

utf8-dot-atom = [CFWS] utf8-dot-atom-text [CFWS]

utf8-dot-atom-text = 1*utf8-atext *("." 1*utf8-atext)

qcontent      = utf8-qcontent
```

To able to use UTF-8 on Content-Description header field on [RFC2045], following syntax is used

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

<utext> syntax is extended on next chapter to allow UTF-8 characters on <unstructured> header fields.

[NOTE IN DRAFT: If any header needs to be restricted to disallow this, please raise the issue on the mailing list.]
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 affects 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 to permit UTF-8 as well as an additional, optional all-ascii alternate address. Note that MSAs and MTAs may downgrade internationalized messages as needed. The procedure for doing so in described in [EAI-downgrading].

```
mailbox        =  name-addr / addr-spec / utf8-addr-spec

angle-addr     =/ [CFWS] "<" utf8-addr-spec [ alt-address ] ">" [CFWS] \
                  / obs-angle-addr

utf8-addr-spec =  utf8-local-part "@" utf8-domain

utf8-local-part=  utf8-dot-atom / utf8-quoted-string / obs-local-part

utf8-domain    =  utf8-dot-atom / domain-literal / obs-domain

alt-address    =  1*FWS "<" addr-spec ">"
```

Below list a few possible <mailbox> representation as example.

```
"DISPLAY_NAME" <ASCII@ASCII>
   ; traditional mailbox format

"DISPLAY_NAME" <non-ASCII@non-ASCII>
   ; UTF8SMTP but no ALT-ADDRESS parameter provided,
   ; message will bounce if UTF8SMTP extension is not supported

<non-ASCII@non-ASCII>
       ; without DISPLAY_NAME and quoted string
       ; UTF8SMTP but no ALT-ADDRESS parameter provided,
   ; message will bounce if UTF8SMTP extension is not supported

"DISPLAY_NAME" <non-ASCII@non-ASCII<ASCII@ASCII>>
   ; UTF8SMTP with ALT-ADDRESS parameter provided,
   ; ALT-ADDRESS can be used if downgrade is necessary
```

4.5. Trace field syntax

"For" fields containing internationalized addresses are allowed, by use of the new uFor syntax. UTF-8 information in needed in Received fields and such information is therefore allowed, to preserve the integrity of those fields. The uFor syntax retains the original UTF-8 email address between EAI-aware MTAs. Note that, should downgrading be required, the uFor parameter is dropped per the procedure specified in [EAI-downgrading].

The "Return-Path" header provides the email return address in the mail delivery. Thus, it MUST able to carry UTF8 addresses (see the revised syntax of <angle-addr> in Section 4.4 of this document). This will not break the rule of trace fied integrity, because it is added at the last MTA.

<item-value> on "Received:" syntax is augmented to allow UTF-8 email address on "For" clause. <angle-addr> is augmented to include UTF-8 email address on previous chapter. To allow UTF-8 email address also on syntax corresponding of <addr-spec> on original syntax, <utf8-addr-spec> is added to <item-value>.

```
item-value      =/      utf8-addr-spec
```

4.6. message/utf8smtp

Internationalized messages, also called UTF8SMTP messages, must only be transmitted as authorized by [EAI-SMTP-extension] or within a non-SMTP environment which supports these messages. A message is a

"UTF8SMTP 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 second type, used for content return, is message/utf8smtp which is similar to message/rfc822, except it contains a message with UTF-8 headers. This type has profound implications on the email infrastructure. First, [RFC3501] servers MUST NOT descend a message/utf8smtp when generating the message BODYSTRUCTURE, it is likely a new variant on BODYSTRUCTURE will be necessary that does descend message/utf8smtp body parts. Second, if this type is sent to a 7-bit-only system, it could be encoded in [RFC2045]. (Note that a system compliant with MIME that doesn't recognize message/utf8smtp would treat it as "application/octet-stream" as described in Section 5.2.4 of [RFC2046].) Alternatively, SMTP servers and other systems which transfer a message/utf8smtp body part MAY choose to down-convert it to a message/rfc822 body part using the rules described in [EAI-downgrading].

Type name: message

Subtype name: utf8smtp

Required parameters: none

Optional parameters: none

Encoding considerations: The 8-bit or binary content-transfer-encoding MUST be used unless this media type is sent over a 7-bit only transport.

Security considerations: See Section 6

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 downconvert the content to message/rfc822. Both of these choices will interoperate with the installed base, but with different properties. Systems unaware of international headers will typically treat a message/utf8smtp body part as an unknown attachment, while they will understand the structure of a message/rfc822. However, systems which understand message/utf8smtp 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 Author's Address section of this document.

Change controller: IETF Standards Process

5. Additional issues

This section identifies issues that are not covered as part of this set of specifications, but that will need to be considered as part of UTF8SMTP deployment.

This document does not specify any requirement for normalization. Prudent use of UTF-8 in identifiers will involve sharply restricted forms, for instance case-folded NFKC, but this document does not require such a form anywhere in the protocol. [Note in draft: Whether this non-requirement is adequate is a subject for debate].

5.1. Mailing list header fields

All mailing list and mail redistribution related headers are discussed in [EAI-mailing-list].

6. 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 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 [RFC2822], 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 which parse, store, or handle email addresses or local parts must take extra care not to overflow buffers, truncate addresses, exceed storage allotments, or, when comparing, fail to use the entire length.

In this specification, a user could provide an ASCII alternative address for a non-ASCII address. However, it is possible these two address go to different mailboxes, or even different persons. This might not be a protocol problem, but instead be the user's personal choice or administration policy or even be a deliberate attempt to deceive or cause confusion.

7. IANA considerations

There are no IANA considerations in this document.

8. Acknowledgements

This document was created by incorporating a good deal of material from an old Internet Draft by Paul Hoffman [Hoffman-utf8-headers]. While many of the concepts and details have changed, the contributions from that draft are greatly appreciated.

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, Chris Newman, Yangwoo KO, Yoshiro

YONEYA, and other members of the JET team and were incorporated into the document. The editor is much great thanks to their contribution sincerely.

9. Edit history

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

9.1. draft-ietf-eai-utf8header-06

1. ABNF revise.
2. Sentences modified
3. Add paragraph in Section 6
4. Add paragraph in Section 1.2
5. Modify Section 4.6

9.2. draft-ietf-eai-utf8header-05

1. ABNF revise.
2. Remove original the section 4 (Pre-requirement)
3. Add UTF8SMTP message (Section 4.6)

9.3. draft-ietf-eai-utf8header-04

1. ABNF revise.
2. Modify uFor description in Section 4.5

9.4. draft-ietf-eai-utf8header-03

1. Editrial changes on terms and english.
2. ABNF revise.
3. addr-spec change, put ALT-ADDRESS inside "<" and ">" quote with "<" and ">".
4. Remove the "Header-Type" header.
5. Add uFor description in Section 4.5
6. Remove the content in IANA considerations since "Header-Type" is removed.

9.5. draft-ietf-eai-utf8header-02

1. Editrial changes on terms and english.
2. Change the header name "UTF8SMTP" to "Header-Type", and ABNF revise.
3. addr-spec change, put ALT-ADDRESS inside "<" and ">" quote with "[" and "]".

4. IANA considerations section rewrite.

9.6. draft-ietf-eai-utf8header-01

1. ABNF revise.
2. Terminology sync with overview document.
3. addr-spec change, put ALT-ADDRESS inside "<" and ">" quote with "{" and "}".
4. add IANA considerations to register the new 2822 header "UTF8SMTP".
5. add Security considerations about relation of UTF8SMTP address to ALT-ADDRESS.

9.7. draft-ietf-eai-utf8header-00

1. ABNF added.
2. Editrial changes.
3. Sent it as WG document.

9.8. draft-yeh-ima-utf8header-01

1. Section re-arranged.
2. Remove content are not below to this document.

10. References

10.1. Normative References

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

ANSI X3.4-1968 has been replaced by newer versions with slight modifications, but the 1968 version remains definitive for the Internet.

[EAI-SMTP-extension] Yao, J., Ed. and Wei. Mao, "SMTP extension for internationalized email address", draft-ietf-eai-smtpext-06.txt (work in progress), June 2007.

[EAI-framework] Klensin, J. and Y. Ko, "Overview and Framework of Internationalized Email Address Delivery", draft-ietf-eai-framework-05.txt (work in progress), Feburary 2007.

[EAI-mailing-list]
Gellens, Randall., "Mailing Lists and Internationalized Email Addresses", draft-ietf-eai-mailinglist-01.txt (work in progress), January 2007.

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

[RFC2821] Klensin, J., "Simple Mail Transfer Protocol", RFC 2821, April 2001.

[RFC2822] Resnick, P., "Internet Message Format", RFC 2822, April 2001.

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

10.2. Informative References

[EAI-downgrading]
YONEYA, Yoshiro., Ed. and Kazunori. Fujiwara, Ed., "Downgrading mechanism for Internationalized eMail Address (IMA)", draft-ietf-eai-downgrade-03.txt (work in progress), March 2007.

[Hoffman-utf8-headers]
Hoffman, P., "SMTP Service Extensions or Transmission of Headers in UTF-8 Encoding", draft-hoffman-utf8headers-00.txt (work in progress), December 2003.

[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.

[RFC3501] Crispin, M., "INTERNET MESSAGE ACCESS PROTOCOL - VERSION 4rev1", RFC 3501, March 2003.

Author's Address

Abel Yang (editor)
TWNIC
4F-2, No. 9, Sec 2, Roosvelt Rd.
Taipei, 100
Taiwan

Phone: +886 2 23411313 ext 505
Email: abelyang@twnic.net.tw

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).