Network Working Group
Request for Comments: 1342

K. Moore
University of Tennessee
June 1992

Representation of Non-ASCII Text in Internet Message Headers

Status of this Memo

This RFC specifies an IAB standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "IAB Official Protocol Standards" for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Abstract

This memo describes an extension to the message format defined in [1] (known to the IETF Mail Extensions Working Group as "RFC 1341"), to allow the representation of character sets other than ASCII in RFC 822 message headers. The extensions described were designed to be highly compatible with existing Internet mail handling software, and to be easily implemented in mail readers that support RFC 1341.

Introduction

RFC 1341 describes a mechanism for denoting textual body parts which are coded in various character sets, as well as methods for encoding such body parts as sequences of printable ASCII characters. This memo describes similar techniques to allow the encoding of non-ASCII text in various portions of a RFC 822 [2] message header, in a manner which is unlikely to confuse existing message handling software.

Like the encoding techniques described in RFC 1341, the techniques outlined here were designed to allow the use of non-ASCII characters in message headers in a way which is unlikely to be disturbed by the quirks of existing Internet mail handling programs. In particular, some mail relaying programs are known to (a) delete some message header fields while retaining others, (b) rearrange the order of addresses in To or Cc fields, (c) rearrange the (vertical) order of header fields, and/or (d) "wrap" message headers at different places than those in the original message. In addition, some mail reading programs are known to have difficulty correctly parsing message headers which, while legal according to RFC 822, make use of backslash-quoting to "hide" special characters such as "<", ",", or or which exploit other infrequently-used features of that specification.

While it is unfortunate that these programs do not correctly interpret RFC 822 headers, to "break" these programs would cause severe operational problems for the Internet mail system. The extensions described in this memo therefore do not rely on little-used features of RFC 822. Instead, certain sequences of "ordinary" printable ASCII characters (which are assumed to be unlikely to otherwise appear in message headers) are reserved for use as encoded data. The characters used in these encodings are restricted to those which do not have special meanings in the context in which the encoded text appears.

Encodings

An "encoded-word" is a sequence of printable ASCII characters that begins with "=?", ends with "?=", and has two "?"s in between. It specifies a character set and an encoding method, and also includes the original text encoded as ASCII characters, according to the rules for that encoding method.

A mail composer that implements this specification will provide a means of inputing non-ASCII text in header fields, but will translate these fields (or appropriate portions of these fields) into encoded-words before inserting them into the message header.

A mail reader that implements this specification will recognize encoded-words when they appear in certain portions of the message header. Instead of displaying the encoded-word "as is", it will reverse the encoding and display the original text in the designated character set.

An "encoded-word" is more precisely defined by the following EBNF grammar, using the notation of RFC 822:

```
encoded-word = "=" "?" charset "?" encoding "?" encoded-text "?" "="

charset = token    ; legal charsets defined by RFC 1341

encoding = token   ; Either "B" or "Q"

token = 1*<Any CHAR except SPACE, CTLs, and tspecials>

tspecials = "(" / ")" / "<" / ">" / "@" / "," / ";" / ":" / "\" /
            <"> / "/" / "[" / "]" / "?" / "." / "="

encoded-text = 1*<Any printable ASCII character other than "?" or
               ; SPACE> (but see "Use of encoded-words in message
               ; headers", below)
```

An encoded-word may not be more than 75 characters long, including charset, encoding, encoded-text, and delimiters. If it is desirable to encode more text than will fit in an encoded-word of 75 characters, multiple encoded-words (separated by SPACE or newline) may be used. Message header lines that contain one or more encoded-words should be no more than 76 characters long. NOTE: These restrictions are included not only to ease interoperbility through internetwork mail gateways, but also to impose a limit on the amount of lookahead a header parser must employ (while looking for a final ?= delimiter) before it can decide whether a token is an encoded-word or something else.

Initially, the legal values for "encoding" are "Q" and "B". These encodings are described below. The "Q" encoding is recommended for use with Latin character sets, and the "B" encoding for all others. Nevertheless, a mail reader which claims to recognize encoded-words MUST be able to accept either encoding for any character set which it supports.

Only a subset of the printable ASCII characters may be used in encoded-text. The SPACE character is not allowed, so that the beginning and end of an encoded-word are obvious. The "?" character is used within an encoded-word to separate the various portions of the encoded-word from one another, and thus cannot appear in the encoded-text portion. Other characters are also illegal in certain contexts. For example, an encoded-word in a "phrase" preceeding an address in a From header field may not contain any of the "specials" defined in RFC 822. Finally, certain other characters are disallowed in some contexts, to ensure reliability for messages that pass through internetwork mail gateways.

The "B" encoding automatically meets these requirements. The "Q" encoding allows a wide range of printable characters to be used in non-critical locations in the message header (e.g., Subject), with fewer characters available for use in other locations.

The "B" encoding

The "B" encoding is identical to the "BASE64" encoding defined by RFC 1341.

The "Q" encoding

The "Q" encoding is similar to the "Quoted-Printable" content-transfer-encoding defined in RFC 1341. It is designed to allow text containing mostly ASCII characters to be decipherable on an ASCII terminal without decoding.

1. Any 8-bit value may be represented by a "=" followed by two hexadecimal digits. For example, if the character set in use were ISO-8859-1, the "=" character would thus be encoded as "=3D", and a SPACE by "=20".

2. The 8-bit hexadecimal value 20 (e.g., IS0-8859-1 SPACE) may be represented as "_" (underscore, ASCII 95.). (This character may not pass through some internetwork mail gateways, but its use will greatly enhance readability of "Q" encoded data with mail readers that do not support this encoding.) Note that the "_" always represents hexadecimal 20, even if the SPACE character occupies a different code position in the character set in use.

3. 8-bit values which correspond to printable ASCII characters other than "=", "?", "_" (underscore), and SPACE may be represented as those characters. (But see "Use of encoded-words in message headers", below).

Character sets

In an encoded-word, the character set associated with the unencoded text is specified by a charset. A charset can be any of the character set names allowed in an RFC 1341 "charset" parameter of a "text/plain" body part. (See section 7.1.1 of RFC 1341 for a list of valid charset parameters).

When there is a possibility of using more than one character set to represent the text in an encoded-word, and in the absence of private agreements between sender and recipients of a message, it is recommended that members of the ISO-8859-* series be used in preference to other character sets. Among the various ISO-8859-* character sets, the lowest-numbered set which contains all of the required characters should be used.

Use of encoded-words in message headers

A sequence of one or more encoded-words is used to represent non-ASCII textual data within a header field. An encoded-word must be separated from an adjacent encoded-word, "word", "text", "ctext", or "special" by a linear white-space character or a newline. When displaying a particular header field" (in the RFC 822 sense) containing one or more encoded-words, an unencoded SPACE character that immediately follows the encoded-word is not displayed. A newline that immediately follows an encoded-word is not displayed unless the encoded-word is the last token in that "field". (This is to allow the use of multiple encoded-words to represent long strings of unencoded text, without having to separate encoded-words where spaces occur in the unencoded text.)

An encoded-word may appear in a message header or body part header according to the following rules:

- An encoded-word may replace a "text" token (as defined by RFC 822) in: (1) a Subject or Comments header field, (2) any extension message header field, (3) any user-defined message header field, or (4) any RFC 1341 body part header field (such as Content-Description) for which the field body contains only "text"s.

- An encoded-word may appear within a comment delimited by "(" and ")", i.e., wherever a "ctext" is allowed. More precisely, the RFC 822 EBNF definition for "comment" is amended as follows:

  comment = "(" *(ctext / quoted-pair / comment / encoded-word) ")"

  A "Q"-encoded encoded-word which appears in a comment MUST NOT contain the characters "(", ")" or "\".

- As a replacement for a "word" entity within a "phrase", for example, one that precedes an address in a From, To, or Cc header. The EBNF definition for phrase from RFC 822 thus becomes:

  phrase = 1*(encoded-word / word)

  In this case the set of characters that may be used in a "Q"-encoded encoded-word is restricted to: <upper and lower case ASCII letters, decimal digits, "!", "*", "+", "-", "/", "=", and "_" (underscore, ASCII 95.)>.

  These are the ONLY locations where an encoded-word may appear. In particular, an encoded-word MUST NOT appear in any portion of an "address". In addition, an encoded-word MUST NOT be used in a Received header field.

  Whenever such words appear in a header being displayed, an enlightened mail reader will decode the text and render it appropriately.

  Only textual data (printable and white space characters) should be encoded using this scheme. However, since these encoding schemes allow the encoding of arbitrary 8-bit values, mail readers that implement this decoding should also ensure that display of the decoded data on the recipient’s terminal will not cause unwanted side-effects.

  Use of these methods to encode non-textual data (e.g., pictures or sounds) is not defined by this memo. Use of encoded-words to represent strings of purely ASCII characters is allowed, but discouraged.

Recognition of encoded-words in message headers.

An encoded-word may be distinguished from an ordinary "word", "text", or "ctext", as follows: An encoded-word begins with "=?", ends with "?=", contains exactly four "?" characters including the delimiters, and is followed by a SPACE or newline. If the "word", "text", or "ctext" does not meet the above tests, it should be displayed as it appears in the message header.

If the mail reader does not support the character set used, it may either display the encoded-word as ordinary text (i.e., as it appears in the header), or it may substitute an appropriate message indicating that the decoded text could not be displayed.

Conformance

A mail composing program claiming compliance with this specification MUST ensure that any string of printable ASCII characters in a message header that begins with "=?" and ends with "?=" be a valid encoded-word.

A mail reading program claiming compliance with this specification must be able to distinguish encoded-words from "text", "ctext", or "word"s anytime they appear in appropriate places in message headers. The program must be able to display unencoded text if the character set is "US-ASCII". For the ISO-8859-* character sets, the mail reading program must at least be able to display the characters which are also in the ASCII set.

Examples

```
From: Keith Moore <moore@cs.utk.edu>
To: Keld Jn Simonsen <keld@dkuug.dk>
CC: Andr Pirard <PIRARD@vm1.ulg.ac.be>
Subject: If you can read this yo=?ISO-8859-2?B?dSB1bmRlcnN0YW5kIHRoZSBleGFtcGxlLg==?=
```

```
From: Olle J舐nefors <ojarnef@admin.kth.se>
To: ietf-822@dimacs.rutgers.edu, ojarnef@admin.kth.se
Subject: Time for ISO 10646?
```

```
To: Dave Crocker <dcrocker@mordor.stanford.edu>
Cc: ietf-822@dimacs.rutgers.edu, paf@comsol.se
From: Patrik F舁tstr <paf@nada.kth.se>
Subject: Re: RFC-HDR care and feeding
```

```
   From: Nathaniel Borenstein <nsb@thumper.bellcore.com>
           (=?iso-8859-8?b?7eXs+SDv4SDp7Oj08A==?=)
   To: Greg Vaudreuil <gvaudre@NRI.Reston.VA.US>, Ned Freed
   <ned@innosoft.com>,
           Keith Moore <moore@cs.utk.edu>
   Subject: Test of new header generator
   MIME-Version: 1.0
   Content-type: text/plain; charset=ISO-8859-1
```

References

[1] Borenstein N., and N. Freed, "MIME (Multipurpose Internet Mail Extensions): Mechanisms for Specifying and Describing the Format of Internet Message Bodies", RFC 1341, Bellcore, Innosoft, June 1992.

[2] Crocker, D., "Standard for the Format of ARPA Internet Text Messages", RFC 822, UDEL, August 1982.

Security Considerations

Security issues are not discussed in this memo.

Author's Address

Keith Moore
University of Tennessee
107 Ayres Hall
Knoxville TN 37996-1301

EMail: moore@cs.utk.edu