The Use of Non-ASCII Characters in RFCs

Abstract

   In order to support the internationalization of protocols and a more
   diverse Internet community, the RFC Series must evolve to allow for
   the use of non-ASCII characters in RFCs. While English remains the
   required language of the Series, the encoding of future RFCs will be
   in UTF-8, allowing for a broader range of characters than typically
   used in the English language. This document describes the RFC Editor
   requirements and gives guidance regarding the use of non-ASCII
   characters in RFCs.

   This document updates RFC 7322. Please view this document in PDF
   form to see the full text.

Table of Contents

1. Introduction

    Please review the PDF version of this draft.

    For much of the history of the RFC Series, the character encoding
    used for RFCs has been ASCII [RFC20]. This was a sensible choice at
    the time: the language of the Series has always been English, a
    language that primarily uses ASCII-encoded characters (ignoring for a
    moment words borrowed from more richly decorated alphabets); and,
    ASCII is the "lowest common denominator" for character encoding,
    making cross-platform viewing trivial.

    There are limits to ASCII, however, that hinder its continued use as
    the exclusive character encoding for the Series. The increasing need
    for easily readable, internationalized content suggests it is time to
    allow non-ASCII characters in RFCs where necessary. To support this
    move away from ASCII, RFCs will switch to supporting UTF-8 as the
    default character encoding and will allow support for a broad range
    of Unicode characters [UnicodeCurrent]. Note that the RFC Editor may
    reject any code point that does not render adequately across all
    formats or in enough rendering engines using the v3 tooling.

    Given the continuing goal of maximum readability across platforms,
    the use of non-ASCII characters should be limited to only where
    necessary within the text. This document describes the rules under
    which non-ASCII characters may be used in an RFC. These rules will
    be applied as the necessary changes are made to submission checking
    and editorial tools.

    This document updates the RFC Style Guide [RFC7322].

    The details included in this document are expected to change based on
    experience gained in implementing the new publication toolsets.
    Revised documents will be published capturing those changes as the
    toolsets are completed. Other implementers must not expect those
    changes to remain backwards compatible with the details included in
    this document.

2. Basic Requirements

    Two fundamental requirements inform the guidance and examples
    provided in this document. They are:

    o Searches against RFC indexes and database tables need to return
       expected results and support appropriate Unicode string matching
       behaviors;

   o RFCs must be able to be displayed correctly across a wide range of
      readers and browsers. People whose systems do not have the fonts
      needed to display a particular RFC need to be able to read the
      various publication formats and the XML correctly in order to
      understand and implement the information described in the
      document.

3. Rules for the Use of Non-ASCII Characters

   This section describes the guidelines for the use of non-ASCII
   characters in an RFC. If the RFC Editor identifies areas where the
   use of non-ASCII characters negatively impacts the readability of the
   text, they will request alternate text.

   The RFC Editor may, in cases of entire words represented in non-ASCII
   characters, ask for a set of reviewers to verify the meaning,
   spelling, characters, and grammar of the text.

3.1. General Usage throughout a Document

   Where the use of non-ASCII characters is purely part of an example
   and not otherwise required for correct protocol operation, escaping
   the non-ASCII character is not required. Note, however, that as the
   language of the RFC Series is English, the use of non-ASCII
   characters is based on the spelling of words commonly used in the
   English language following the guidance in the Merriam-Webster
   dictionary [MerrWeb].

   The RFC Editor will use the primary spelling listed in that
   dictionary by default.

   Example of non-ASCII characters that do not require escaping (example
   from Section 3.1.1.12 of RFC 4475 [RFC4475], with a hex dump replaced
   by the actual character glyphs):

   This particular response contains unreserved and non-ASCII
   UTF-8 characters.
   This response is well formed. A parser must accept this message.

   Message Details : unreason

   SIP/2.0 200 = 2**3 * 5**2 (See PDF for non-ASCII character string)
   Via: SIP/2.0/UDP 192.0.2.198;branch=z9hG4bK1324923
   Call-ID: unreason.1234ksdfak3j2erwedfsASdf
   CSeq: 35 INVITE
   From: sip:user@example.com;tag=11141343
   To: sip:user@example.edu;tag=2229 Content-Length: 154
   Content-Type: application/sdp

3.2. Person Names

   Person names may appear in several places within an RFC (e.g., the
   header, Acknowledgements, and References). When a script outside the
   Unicode Latin blocks [UNICODE-CHART] is used for an individual name,
   an author-provided, ASCII-only identifier will appear immediately
   after the non-Latin characters, surrounded by parentheses. This will
   improve general readability of the text.

   Example header:

   OLD:

   Internet Engineering Task Force (IETF)   J. Tong
   Request for Comments: 7380   C. Bi, Ed.
   Category: Standards Track   China Telecom
   ISSN: 2070-1721   R. Even
                                                             Q. Wu, Ed.
                                                              R. Huang
                                                                Huawei
                                                          November 2014

   PROPOSED/NEW:

   Internet Engineering Task Force (IETF)   J. Tong
   Request for Comments: 7380   C. Bi, Ed.
   Category: Standards Track   China Telecom
   ISSN: 2070-1721   (See PDF for non-ASCII character string) (R. Even)
                  (See PDF for non-ASCII character string) (Q. Wu), Ed.
                                                              R. Huang
                                                                Huawei
                                                          November 2014

Example Acknowledgements section:

OLD:

The following people contributed significant text to early versions
of this draft: Patrik Faltstrom, William Chan, and Fred Baker.

PROPOSED/NEW:

The following people contributed significant text to early versions
of this draft: Patrik (See PDF for non-ASCII character string)
(Faltstrom), (See PDF for non-ASCII character string) (William Chan),
and Fred Baker.

Example reference entry:

OLD:

    [RFC6630] Cao, Z., Deng, H., Wu, Q., and G. Zorn, Ed., "EAP
               Re-authentication Protocol Extensions for Authenticated
               Anticipatory Keying (ERP/AAK)", RFC 6630, June 2012.

NEW

    [RFC6630] Cao, Z., Deng, H., (See PDF for non-ASCII character
               string) (Wu, Q.), and G. Zorn, Ed., "EAP
               Re-authentication Protocol Extensions for Authenticated
               Anticipatory Keying (ERP/AAK)", RFC 6630, June 2012.

3.3. Company Names

   Company names may appear in several places within an RFC. In all
   cases, valid Unicode is required. For names that include characters
   outside of the Unicode Latin and Latin Extended scripts, an author-
   provided, ASCII-only identifier is required to assist in searching
   and indexing of the document.

3.4. Body of the Document

   When the mention of non-ASCII characters is required for correct
   protocol operation and understanding, the characters' Unicode code
   points must be used in the text. The addition of each character name
   is encouraged.

   o Non-ASCII characters will require identifying the Unicode code
      point.

   o Use of the actual UTF-8 character (e.g., (See PDF for non-ASCII
      character string)) is encouraged so that a reader can more easily
      see what the character is, if their device can render the text.

   o The use of the Unicode character names like "INCREMENT" in
      addition to the use of Unicode code points is also encouraged.
      When used, Unicode character names should be in all capital
      letters.

   Examples:

   OLD [RFC7564]:

   However, the problem is made more serious by introducing the full
   range of Unicode code points into protocol strings. For example,
   the characters U+13DA U+13A2 U+13B5 U+13AC U+13A2 U+13AC U+13D2 from
   the Cherokee block look similar to the ASCII characters
   "STPETER" as they might appear when presented using a "creative"
   font family.

   NEW/ALLOWED:

   However, the problem is made more serious by introducing the full
   range of Unicode code points into protocol strings. For example,
   the characters U+13DA U+13A2 U+13B5 U+13AC U+13A2 U+13AC U+13D2
   ((See PDF for non-ASCII character string)) from the Cherokee
   block look similar to the ASCII characters "STPETER" as they might
   appear when presented using a "creative" font family.

   ALSO ACCEPTABLE:

   However, the problem is made more serious by introducing the full
   range of Unicode code points into protocol strings. For example,
   the characters "(See PDF for non-ASCII character string)" (U+13DA
   U+13A2 U+13B5 U+13AC U+13A2 U+13AC U+13D2) from the Cherokee block
   look similar to the ASCII characters "STPETER" as they might
   appear when presented using a "creative" font family.

Example of proper identification of Unicode characters in an RFC:

Acceptable:

o Temperature changes in the Temperature Control Protocol are
   indicated by the U+2206 character.

Preferred:

1. Temperature changes in the Temperature Control Protocol are
    indicated by the U+2206 character ("(See PDF for non-ASCII
    character string)").

2. Temperature changes in the Temperature Control Protocol are
    indicated by the U+2206 character (INCREMENT).

3. Temperature changes in the Temperature Control Protocol are
    indicated by the U+2206 character ("(See PDF for non-ASCII
    character string)", INCREMENT).

4. Temperature changes in the Temperature Control Protocol are
    indicated by the U+2206 character (INCREMENT, "(See PDF for non-
    ASCII character string)").

5. Temperature changes in the Temperature Control Protocol are
    indicated by the "Delta" character "(See PDF for non-ASCII
    character string)" (U+2206).

6. Temperature changes in the Temperature Control Protocol are
    indicated by the character "(See PDF for non-ASCII character
    string)" (INCREMENT, U+2206).

Which option of (1), (2), (3), (4), (5), or (6) is preferred may
depend on context and the specific character(s) in question. All are
acceptable within an RFC. "US-ASCII Escaping of Unicode Character"
[BCP137] describes the pros and cons of different options for
identifying Unicode characters and may help authors decide how to
represent the non-ASCII characters in their documents.

3.5. Tables

   Tables follow the same rules for identifiers and characters as in
   "Body of the Document" (Section 3.4). If it is sensible (i.e., more
   understandable for a reader) for a given document to have two tables,
   -- one including the identifiers and non-ASCII characters and a
   second with just the non-ASCII characters -- then that will be
   allowed at the discretion of the authors.

   Original text from "Preparation, Enforcement, and Comparison of
   Internationalized Strings Representing Usernames and Passwords"
   [RFC7613].

Table 3: A sample of legal passwords

```
+----------------------------------+------------------------------+
| # | Password                     | Notes                        |
+----------------------------------+------------------------------+
| 12| <correct horse battery staple> | ASCII space is allowed     |
+----------------------------------+------------------------------+
| 13| <Correct Horse Battery Staple> | Different from example 12  |
+----------------------------------+------------------------------+
| 14| <&#x3C0;&#xDF;&#xE5;>         | Non-ASCII letters are OK     |
|   |                              | (e.g., GREEK SMALL LETTER    |
|   |                              | PI, U+03C0)                  |
+----------------------------------+------------------------------+
| 15| <Jack of &#x2666;s>          | Symbols are OK (e.g., BLACK  |
|   |                              | DIAMOND SUIT, U+2666)        |
+----------------------------------+------------------------------+
| 16| <foo&#x1680;bar>             | OGHAM SPACE MARK, U+1680, is |
|   |                              | mapped to U+0020 and thus    |
|   |                              | the full string is mapped to |
|   |                              | <foo bar>                    |
+----------------------------------+------------------------------+
```

Preferred text:

Table 3: A sample of legal passwords

```
+----------------------------------+------------------------------+
| # | Password                     | Notes                        |
+----------------------------------+------------------------------+
| 12| <correct horse battery staple> | ASCII space is allowed     |
+----------------------------------+------------------------------+
| 13| <Correct Horse Battery Staple> | Different from example 12  |
+----------------------------------+------------------------------+
| 14| <(See PDF for non-ASCII      | Non-ASCII letters are OK     |
|   |    character string)>        | (e.g., GREEK SMALL LETTER    |
|   |                              | PI, U+03C0; LATIN SMALL      |
|   |                              | LETTER SHARP S, U+00DF; THAI |
|   |                              | DIGIT SEVEN, U+0E57)         |
+----------------------------------+------------------------------+
| 15| <Jack of (See PDF for non-   | Symbols are OK (e.g., BLACK  |
|   |    ASCII character string)s>  | DIAMOND SUIT, U+2666)       |
+----------------------------------+------------------------------+
| 16| <foo(See PDF for non-ASCII   | OGHAM SPACE MARK, U+1680, is |
|   |    character string)bar>     | mapped to U+0020 and thus    |
|   |                              | the full string is mapped to |
|   |                              | <foo bar>                    |
+----------------------------------+------------------------------+
```

3.6. Code Components

   The RFC Editor encourages the use of the U+ notation except within a
   code component where one must follow the rules of the programming
   language in which the code is being written.

   Code components are generally expected to use fixed-width fonts.
   Where such fonts are not available for a particular script, the best
   script-appropriate font will be used for that part of the code
   component.

3.7. Bibliographic Text

   The reference entry must be in English; whatever subfields are
   present must be available in ASCII-encoded characters. For
   references to RFCs and Internet-Drafts, the author's name will be
   formatted in the reference as per current RFC Style Guide
   recommendations. As long as good sense is used, the reference entry
   may also include non-ASCII characters at the author's discretion and
   as provided by the author. The RFC Editor may request that a third
   party, such as a language specialist or subject matter expert, review
   of any non-ASCII reference. This applies to both normative and
   informative references.

   Example:

   [GOST3410] "Information technology. Cryptographic data security.
           Signature and verification processes of [electronic]
           digital signature.", GOST R 34.10-2001, Gosudarstvennyi
           Standard of Russian Federation, Government Committee of
           Russia for Standards, 2001. (In Russian)

   Allowable addition to the above citation:
           (See PDF for non-ASCII character strings)

   Alternatively:
   [GOST3410] "Information technology. Cryptographic data security.
           Signature and verification processes of [electronic]
           digital signature.", GOST R 34.10-2001, Gosudarstvennyi
           Standard of Russian Federation, (See PDF for non-ASCII
           character strings) (Government Committee of
           Russia for Standards), 2001. (In Russian)

3.8. Keywords and Citation Tags

   Keywords (as tagged with the <keyword> element in XML) and citation
   tags (as defined in the anchor attributes of <reference> elements)
   must contain only ASCII characters.

3.9. Address Information

   The purpose of providing address information, either postal or email,
   is to assist readers of an RFC in contacting the author or authors.
   Authors may include the official postal address as recognized by
   their company or local postal service without additional non-ASCII
   character escapes. If the email address includes non-ASCII
   characters and is a valid email address at the time of publication,
   non-ASCII character escapes are not required.

   Example:

     Qin Wu (editor)
     Huawei
     101 Software Avenue, Yuhua District
     Nanjing, Jiangsu 210012
     China

   Additional contact information:
     (See PDF for non-ASCII character strings)

     Roni Even
     Huawei
     14 David Hamelech
     Tel Aviv 64953
     Israel

   Additional contact information:
      (See PDF for non-ASCII character strings)

4. Normalization Forms

   Authors should not expect normalization forms [UNICODE-NORM]to be
   preserved. If a particular normalization form is expected, note that
   in the text of the RFC.

5. XML Markup

   As described above, use of non-ASCII characters in areas such as
   email, company name, address, and name is allowed. In order to make
   it easier for code to identify the appropriate ASCII alternatives,
   authors must include an "ascii" attribute to their XML markup when an
   ASCII alternative is required. See [RFC7991] for more detail on how
   to tag ASCII alternatives.

6. Internationalization Considerations

   The ability to use non-ASCII characters in RFCs in a clear and
   consistent manner will improve the ability to describe
   internationalized protocols and will recognize the diversity of
   authors. However, the goal of readability will override the use of
   non-ASCII characters within the text.

7. Security Considerations

   Valid Unicode that matches the expected text must be verified in
   order to preserve expected behavior and protocol information.

8. Informative References

   [BCP137]   Klensin, J., "ASCII Escaping of Unicode Characters",
              BCP 137, RFC 5137, February 2008,
              <http://www.rfc-editor.org/info/bcp137>.

   [MerrWeb] Merriam-Webster, Inc., "Merriam-Webster's Collegiate
              Dictionary, 11th Edition", 2009.

   [RFC20]   Cerf, V., "ASCII format for network interchange", STD 80,
              RFC 20, DOI 10.17487/RFC0020, October 1969,
              <http://www.rfc-editor.org/info/rfc20>.

   [RFC4475] Sparks, R., Ed., Hawrylyshen, A., Johnston, A., Rosenberg,
              J., and H. Schulzrinne, "Session Initiation Protocol (SIP)
              Torture Test Messages", RFC 4475, DOI 10.17487/RFC4475,
              May 2006, <http://www.rfc-editor.org/info/rfc4475>.

   [RFC7322] Flanagan, H. and S. Ginoza, "RFC Style Guide", RFC 7322,
              DOI 10.17487/RFC7322, September 2014,
              <http://www.rfc-editor.org/info/rfc7322>.

   [RFC7564] Saint-Andre, P. and M. Blanchet, "PRECIS Framework:
              Preparation, Enforcement, and Comparison of
              Internationalized Strings in Application Protocols",
              RFC 7564, DOI 10.17487/RFC7564, May 2015,
              <http://www.rfc-editor.org/info/rfc7564>.

   [RFC7613] Saint-Andre, P. and A. Melnikov, "Preparation,
              Enforcement, and Comparison of Internationalized Strings
              Representing Usernames and Passwords", RFC 7613,
              DOI 10.17487/RFC7613, August 2015,
              <http://www.rfc-editor.org/info/rfc7613>.

   [RFC7991]   Hoffman, P., "The "xml2rfc" Version 3 Vocabulary",
               RFC 7991, DOI 10.17487/RFC7991, December 2016,
               <http://www.rfc-editor.org/info/rfc7991>.

   [UNICODE-CHART]
               The Unicode Consortium, "The Unicode Standard",
               <http://www.unicode.org/charts>.

   [UNICODE-NORM]
               The Unicode Consortium, "Unicode Standard Annex #15:
               Unicode Normalization Forms", 2016,
               <http://www.unicode.org/reports/tr15/>.

   [UnicodeCurrent]
               The Unicode Consortium, "The Unicode Standard",
               <http://www.unicode.org/versions/latest/>.

IAB Members at the Time of Approval

   The IAB members at the time this memo was approved were (in
   alphabetical order):

      Jari Arkko
      Ralph Droms
      Ted Hardie
      Joe Hildebrand
      Russ Housley
      Lee Howard
      Erik Nordmark
      Robert Sparks
      Andrew Sullivan
      Dave Thaler
      Martin Thomson
      Brian Trammell
      Suzanne Woolf

Acknowledgements

   With many thanks to the members of the IAB i18n program. Also, many
   thanks to the RFC Format Design Team for their efforts in making this
   transition successful: Nevil Brownlee (ISE), Tony Hansen, Joe
   Hildebrand, Paul Hoffman, Ted Lemon, Julian Reschke, Adam Roach,
   Alice Russo, Robert Sparks (Tools Team liaison), and Dave Thaler.

Author's Address

    Heather Flanagan (editor)
    RFC Editor

    Email: rse@rfc-editor.org
    URI: http://orcid.org/0000-0002-2647-2220