Network Working Group H. Flanagan
Internet-Draft RFC Editor
Intended status: Informational April 18, 2014
Expires: October 20, 2014

The Use of Non-ASCII Characters in RFCs
draft-flanagan-nonascii-01

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 accepted language of the Series, the encoding of future RFCs will be in UTF-8. This document describes the RFC Editor requirements and guidance regarding the use of non-ASCII characters in RFCs.

This document updates [draft-iab-styleguide].

Please review the PDF version of this draft.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

For much of the history of the RFC Series, the character encoding used for RFCs has been ASCII [ASCII]. This was a sensible choice at the time: the language of the Series is English, a language that only 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 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 [STD63]. UTF-8 has seen widespread acceptance by authors, publishers, and code developers across the Internet, is backwards-compatible with ASCII, and is the default encoding for XML (the new canonical format of RFCs) [RFC6949].

Given the continuing goal of maximum readability across platforms, the use of non-ASCII characters should be limited in a document 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.

2. Basic requirements

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

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

- RFCs must be able to display correctly across a wide range of readers and browsers. People whose system does not have the fonts needed to display a particular RFC need to be able to read the non-canonical HTML, text, or PDF RFC correctly.

3. Rules for the use of non-ASCII characters

This section describes the guidelines for the use of non-ASCII characters in the header, body, and reference sections of 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 as part of an example and not otherwise required for correct protocol operation, escaping the Unicode 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 the dictionary by default.

Example of non-ASCII characters that do not require escaping [RFC4475]:

> 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 но сто девяносто девять - простое
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. Authors, Contributors, and Acknowledgments

Person names may appear in several places within an RFC. In all cases, valid Unicode is required. For names that include non-ASCII characters, an author-provided, ASCII-only identifier is required to assist in search and indexing of the document.

Example for the header:

```
Network Working Group                                      L. Daigle
Request for Comments: 2611             Thinking Cat Enterprises
BCP: 33                                                 D. van Gulik
Category: Best Current Practice               ISIS/CEO, JRC Ispra
                                                         R. Iannella
                                                        DSTC Pty Ltd
                                   P. Fältström (P. Faltstrom)
                                                       Tele2/Swipnet
                                                           June 1999
```

Example for the Acknowledgements:

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 Fältström (Patrik Faltstrom), 陈智昌 (William Chan), and Fred Baker.

3.3. Company Names

Company names may appear in several places within an RFC. The rules for company names follow similar guidance to that of person names. Valid Unicode is required. For company names that include non-ASCII characters, an ASCII-only identifier is required to assist in search 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 character name or code point MUST be included in the text.

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

- Use of the actual UTF-8 character (e.g., ∆) is encouraged so that a reader can more easily see what the character is, if their device can render the text.

- 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 [draft-ietf-precis-framework]:
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 (ᏚᎢᎵᎬᎢᎬᏒ) 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 "ᏚᎢᎵᎬᎢᎬᏒ" (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:

Temperature changes in the Temperature Control Protocol are indicated by the U+0394 character.

Preferred:

(a) Temperature changes in the Temperature Control Protocol are indicated by the U+2206 character (“∆”).

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

(c) Temperature changes in the Temperature Control Protocol are indicated by the U+2206 character (“∆”, INCREMENT).

(d) Temperature changes in the Temperature Control Protocol are indicated by the U+2206 character (INCREMENT, “∆”).

(e) Temperature changes in the Temperature Control Protocol are indicated by the [Delta] character "∆" (U+2206).

(f) Temperature changes in the Temperature Control Protocol are indicated by the character "∆" (INCREMENT, U+2206).

Which option of (a), (b), (c), (d), (e), or (f) is preferred may depend on context and the specific character(s) in question. All are acceptable within an RFC. BCP 137, “ASCII Escaping of Unicode Character” describes the pros and cons of different options for identifying Unicode characters in an ASCII document [BCP137].

3.5. Tables

Tables follow the same rules for identifiers and characters as in “Section 3.4 Body of the document”. 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 -- that will be allowed on a case-by-case basis.

Example: TBD

3.6. Code components

The RFC Editor encourages the use of the U+ notation except within a code component where you must follow the rules of the programming language in which you are writing the code.

Example:

TBD

3.7. Bibliographic text

The reference entry must be in English; whatever subfields are present must be available in ASCII-encoded characters. 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 will request a review of the non-ASCII reference entry.

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:

"**Информационная технология. Криптографическая защита информации. Процессы формирования и проверки электронной цифровой подписи** ", GOST R 34.10-2001, **Государственный стандарт Российской Федерации**, 2001.

3.8. Keywords

Keywords must be ASCII only.

3.9. Address Information

The purpose of providing address information, either postal or e-mail, is to assist readers of an RFC to contact 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.

4. Normalization Forms

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

5. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

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.

7. Security Considerations

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

8. References

[ASCII] American National Standard for Information Systems – Coded Character Sets - 7-Bit American National Standard Code for Information Interchange (7-Bit ASCII), ANSI X3.4- 1986, American National Standards Institute, Inc., March 26, 1986.

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

[STD63] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003, <http://www.rfc-editor.org/std/std63.txt>.

[RFC6949] Flanagan, H. and N. Brownlee, "RFC Series Format Requirements and Future Development", RFC 6949, May 2013, <http://www.rfc-editor.org/info/rfc6949>.

Acknowledgements

With many thanks to the members of the IAB i18n program.

Author's Address

Heather Flanagan
RFC Editor

Email: rse@rfc-editor.org