          Preparation, Enforcement, and Comparison of Internationalized Strings
                          Representing Nicknames
                        draft-ietf-precis-7700bis-03

Abstract

   This document describes methods for handling Unicode strings
   representing memorable, human-friendly names (called "nicknames",
   "display names", or "petnames") for people, devices, accounts,
   websites, and other entities. This document obsoletes RFC 7700.

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 9, 2017.

Table of Contents

1. Introduction

1.1. Overview

   A number of technologies and applications provide the ability for a
   person to choose a memorable, human-friendly name in a communications
   context, or to set such a name for another entity such as a device,
   account, contact, or website. Such names are variously called
   "nicknames" (e.g., in chat room applications), "display names" (e.g.,
   in Internet mail), or "petnames" (see [PETNAME-SYSTEMS]); for
   consistency, these are all called "nicknames" in this document.

   Nicknames are commonly supported in technologies for textual chat
   rooms, e.g., Internet Relay Chat [RFC2811] and multi-party chat
   technologies based on the Extensible Messaging and Presence Protocol
   (XMPP) [RFC6120] [XEP-0045], the Message Session Relay Protocol
   (MSRP) [RFC4975] [RFC7701], and Centralized Conferencing (XCON)
   [RFC5239] [XCON-SYSTEM]. Recent chat room technologies also allow
   internationalized nicknames because they support characters from
   outside the ASCII range [RFC20], typically by means of the Unicode
   character set [Unicode]. Although such nicknames tend to be used

primarily for display purposes, they are sometimes used for
programmatic purposes as well (e.g., kicking users or avoiding
nickname conflicts).

A similar usage enables a person to set their own preferred display
name or to set a preferred display name for another user (e.g., the
"display-name" construct in the Internet message format [RFC5322] and
[XEP-0172] in XMPP).

Memorable, human-friendly names are also used in contexts other than
personal messaging, such as names for devices (e.g., in a network
visualization application), websites (e.g., for bookmarks in a web
browser), accounts (e.g., in a web interface for a list of payees in
a bank account), people (e.g., in a contact list application), and
the like.

The rules specified in this document can be applied in all of the
foregoing contexts.

To increase the likelihood that memorable, human-friendly names will
work in ways that make sense for typical users throughout the world,
this document defines rules for preparing, enforcing, and comparing
internationalized nicknames.

1.2. Terminology

Many important terms used in this document are defined in [RFC7564],
[RFC6365], and [Unicode].

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

2. Nickname Profile

2.1. Rules

The following rules apply within the Nickname profile of the PRECIS
FreeformClass.

1. Width Mapping Rule: There is no width-mapping rule (such a rule
    is not necessary because width mapping is performed as part of
    normalization using Normalization Form KC (NFKC) as specified
    below).

2. Additional Mapping Rule: The additional mapping rule consists of
    the following sub-rules.

   1. Map any instances of non-ASCII space to ASCII space (U+0020);
       a non-ASCII space is any Unicode code point having a general
       category of "Zs", naturally with the exception of U+0020.

   2. Remove any instances of the ASCII space character at the
       beginning or end of a nickname (e.g., "stpeter " is mapped to
       "stpeter").

   3. Map interior sequences of more than one ASCII space character
       to a single ASCII space character (e.g., "St Peter" is
       mapped to "St Peter").

3. Case Mapping Rule: Apply the Unicode toLower() operation, as
    defined in the Unicode Standard [Unicode]. In applications that
    prohibit conflicting nicknames, this rule helps to reduce the
    possibility of confusion by ensuring that nicknames differing
    only by case (e.g., "stpeter" vs. "StPeter") would not be
    presented to a human user at the same time. (As explained below,
    this is typically appropriate only for comparison, not for
    enforcement.)

4. Normalization Rule: Apply Unicode Normalization Form KC. Because
    NFKC is more "aggressive" in finding matches than other
    normalization forms (in the terminology of Unicode, it performs
    both canonical and compatibility decomposition before recomposing
    code points), this rule helps to reduce the possibility of
    confusion by increasing the number of characters that would match
    (e.g., U+2163 ROMAN NUMERAL FOUR would match the combination of
    U+0049 LATIN CAPITAL LETTER I and U+0056 LATIN CAPITAL LETTER V).

5. Directionality Rule: There is no directionality rule. The "Bidi
    Rule" (defined in [RFC5893]) and similar rules are unnecessary
    and inapplicable to nicknames, because it is perfectly acceptable
    for a given nickname to be presented differently in different
    layout systems (e.g., a user interface that is configured to
    handle primarily a right-to-left script versus an interface that
    is configured to handle primarily a left-to-right script), as
    long as the presentation is consistent in any given layout
    system.

2.2. Preparation

   An entity that prepares a string for subsequent enforcement according
   to this profile MUST ensure that the string consists only of Unicode
   code points that conform to the FreeformClass string class defined in
   [RFC7564].

2.3. Enforcement

An entity that performs enforcement according to this profile MUST
prepare a string as described in Section 2.2 and MUST also apply the
following rules specified in Section 2.1 in the order shown:

1.   Additional Mapping Rule
2.   Normalization Rule

Note: An entity SHOULD NOT apply the Case Mapping Rule during
enforcement, because typically it is appropriate only during
comparison.

After all of the foregoing rules have been enforced, the entity MUST
ensure that the nickname is not zero bytes in length (this is done
after enforcing the rules to prevent applications from mistakenly
omitting a nickname entirely, because when internationalized
characters are accepted, a non-empty sequence of characters can
result in a zero-length nickname after canonicalization).

2.4. Comparison

An entity that performs comparison of two strings according to this
profile MUST prepare each string as specified in Section 2.2 and MUST
apply the following rules specified in Section 2.1 in the order
shown:

1.   Additional Mapping Rule
2.   Case Mapping Rule
3.   Normalization Rule

The two strings are to be considered equivalent if they are an exact
octet-for-octet match (sometimes called "bit-string identity").

3. Examples

The following examples illustrate a small number of nicknames that
are consistent with the format defined above, along with the output
string resulting from application of the PRECIS rules (note that the
characters < and > are used to delineate the actual nickname and are
not part of the nickname strings).

Table 1: A Sample of Legal Nicknames

```
+---------------------------+-----------------------------------+
| # | Nickname              | Output for Comparison             |
+---------------------------+-----------------------------------+
| 1 | <Foo>                 | <foo>                             |
+---------------------------+-----------------------------------+
| 2 | <foo>                 | <foo>                             |
+---------------------------+-----------------------------------+
| 3 | <Foo Bar>             | <foo bar>                         |
+---------------------------+-----------------------------------+
| 4 | <foo bar>             | <foo bar>                         |
+---------------------------+-----------------------------------+
| 5 | <&#x3A3;>             | GREEK SMALL LETTER SIGMA (U+03C3) |
+---------------------------+-----------------------------------+
| 6 | <&#x3C3;>             | GREEK SMALL LETTER SIGMA (U+03C3) |
+---------------------------+-----------------------------------+
| 7 | <&#x3C2;>             | GREEK SMALL LETTER FINAL SIGMA    |
|   |                       | (U+03C2) |                        |
+---------------------------+-----------------------------------+
| 8 | <&#x265A;>            | BLACK CHESS KING (U+265A)         |
+---------------------------+-----------------------------------+
| 9 | <Richard &#x2163;>    | <richard iv>                      |
+---------------------------+-----------------------------------+
```

Regarding examples 5, 6, and 7: applying the Unicode toLower()
operation to GREEK CAPITAL LETTER SIGMA (U+03A3) results in GREEK
SMALL LETTER SIGMA (U+03C3), however the toLower() operation does not
modify GREEK SMALL LETTER FINAL SIGMA (U+03C2); therefore, the
comparison operation defined in [Section 2.4](#) would result in matching
of the nicknames in examples 5 and 6 but not the nicknames in
examples 5 and 7 or 6 and 7. Regarding example 8: symbol characters
such as BLACK CHESS KING (U+265A) are allowed by the PRECIS
FreeformClass and thus can be used in nicknames. Regarding example
9: applying the Unicode toLower() operation to ROMAN NUMERAL FOUR
(U+2163) results in SMALL ROMAN NUMERAL FOUR (U+2173), and applying
NFKC to SMALL ROMAN NUMERAL FOUR (U+2173) results in LATIN SMALL
LETTER I (U+0069) LATIN SMALL LETTER V (U+0086).

[4](#). Use in Application Protocols

This specification defines only the PRECIS-based rules for handling
of nickname strings. It is the responsibility of an application
protocol (e.g., MSRP, XCON, or XMPP) or application definition to
specify the protocol slots in which nickname strings can appear, the
entities that are expected to enforce the rules governing nickname
strings, and when in protocol processing or interface handling the

rules need to be enforced. See Section 6 of [RFC7564] for guidelines
about using PRECIS profiles in applications.

Above and beyond the PRECIS-based rules specified here, application
protocols can also define application-specific rules governing
nickname strings (rules regarding the minimum or maximum length of
nicknames, further restrictions on allowable characters or character
ranges, safeguards to mitigate the effects of visually similar
characters, etc.).

Naturally, application protocols can also specify rules governing the
actual use of nicknames in applications (reserved nicknames,
authorization requirements for using nicknames, whether certain
nicknames can be prohibited, handling of duplicates, the relationship
between nicknames and underlying identifiers such as SIP URIs or
Jabber IDs, etc.).

Entities that enforce the rules specified in this document are
encouraged to be liberal in what they accept by following this
procedure:

1. Where possible, map characters (e.g, through width mapping,
    additional mapping, case mapping, or normalization) and accept
    the mapped string.

2. If mapping is not possible (e.g., because a character is
    disallowed in the FreeformClass), reject the string.

5. IANA Considerations

The IANA shall add the following entry to the PRECIS Profiles
Registry:

Name: Nickname

Base Class: FreeformClass

Applicability: Nicknames in messaging and text conferencing
   technologies; petnames for devices, accounts, and people; and
   other uses of nicknames or petnames.

Replaces: None

Width Mapping Rule: None (handled via NFKC)

Additional Mapping Rule: Map non-ASCII space characters to ASCII
   space, strip leading and trailing space characters, map interior
   sequences of multiple space characters to a single ASCII space.

   Case Mapping Rule: Map uppercase and titlecase characters to
      lowercase using the Unicode toLower() operation.

   Normalization Rule: NFKC

   Directionality Rule: None

   Enforcement: To be specified by applications.

   Specification: RFC 7700 (this document)

6. Security Considerations

6.1. Reuse of PRECIS

   The security considerations described in [RFC7564] apply to the
   FreeformClass string class used in this document for nicknames.

6.2. Reuse of Unicode

   The security considerations described in [UTS39] apply to the use of
   Unicode characters in nicknames.

6.3. Visually Similar Characters

   [RFC7564] describes some of the security considerations related to
   visually similar characters, also called "confusable characters" or
   "confusables".

   Although the mapping rules defined in Section 2 of this document are
   designed, in part, to reduce the possibility of confusion about
   nicknames, this document does not provide more-detailed
   recommendations regarding the handling of visually similar
   characters, such as those provided in [UTS39].

7. References

7.1. Normative References

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

   [RFC5893] Alvestrand, H., Ed. and C. Karp, "Right-to-Left Scripts
              for Internationalized Domain Names for Applications
              (IDNA)", RFC 5893, DOI 10.17487/RFC5893, August 2010,
              <http://www.rfc-editor.org/info/rfc5893>.

   [RFC6365] Hoffman, P. and J. Klensin, "Terminology Used in
              Internationalization in the IETF", BCP 166, RFC 6365, DOI
              10.17487/RFC6365, September 2011,
              <http://www.rfc-editor.org/info/rfc6365>.

   [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>.

   [UTS39]   The Unicode Consortium, "Unicode Technical Standard #39:
              Unicode Security Mechanisms", November 2013,
              <http://unicode.org/reports/tr39/>.

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

7.2. Informative References

   [Err4570] RFC Errata, "Erratum ID 4570", RFC 7700,
              <http://www.rfc-editor.org>.

   [PETNAME-SYSTEMS]
              Stiegler, M., "An Introduction to Petname Systems",
              updated June 2012, February 2005,
              <http://www.skyhunter.com/marcs/petnames/
              IntroPetNames.html>.

   [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>.

   [RFC2811] Kalt, C., "Internet Relay Chat: Channel Management", RFC
              2811, DOI 10.17487/RFC2811, April 2000,
              <http://www.rfc-editor.org/info/rfc2811>.

   [RFC4975] Campbell, B., Ed., Mahy, R., Ed., and C. Jennings, Ed.,
              "The Message Session Relay Protocol (MSRP)", RFC 4975, DOI
              10.17487/RFC4975, September 2007,
              <http://www.rfc-editor.org/info/rfc4975>.

   [RFC5239] Barnes, M., Boulton, C., and O. Levin, "A Framework for
              Centralized Conferencing", RFC 5239, DOI 10.17487/RFC5239,
              June 2008, <http://www.rfc-editor.org/info/rfc5239>.

   [RFC5322] Resnick, P., Ed., "Internet Message Format", RFC 5322, DOI
              10.17487/RFC5322, October 2008,
              <http://www.rfc-editor.org/info/rfc5322>.

   [RFC6120] Saint-Andre, P., "Extensible Messaging and Presence
              Protocol (XMPP): Core", RFC 6120, March 2011.

   [RFC7700] Saint-Andre, P., "Preparation, Enforcement, and Comparison
              of Internationalized Strings Representing Nicknames", RFC
              7700, DOI 10.17487/RFC7700, December 2015,
              <http://www.rfc-editor.org/info/rfc7700>.

   [RFC7701] Niemi, A., Garcia-Martin, M., and G. Sandbakken, "Multi-
              party Chat Using the Message Session Relay Protocol
              (MSRP)", RFC 7701, DOI 10.17487/RFC7701, December 2015,
              <http://www.rfc-editor.org/info/rfc7701>.

   [XCON-SYSTEM]
              Barnes, M., Boulton, C., and S. Loreto, "Chatrooms within
              a Centralized Conferencing (XCON) System", Work in
              Progress, draft-boulton-xcon-session-chat-08, July 2012.

   [XEP-0045]
              Saint-Andre, P., "Multi-User Chat", XSF XEP 0045, February
              2012, <http://xmpp.org/extensions/xep-0045.html>.

   [XEP-0172]
              Saint-Andre, P. and V. Mercier, "User Nickname", XSF XEP
              0172, March 2012,
              <http://xmpp.org/extensions/xep-0172.html>.

Appendix A. Differences from RFC 7700

   The following modifications were made from [RFC7700].

   o Addressed [Err4570] by removing the directionality rule and adding
      the normalization rule to Section 2.3.

   o In accordance with working group discussions and updates to
      [RFC7564], removed the use of the Unicode CaseFold() operation in
      favor of the Unicode toLower() operation.

   o Clarified several editorial matters.

   o Updated references.

Appendix B. Acknowledgements

Thanks to Sam Whited for his feedback and for submitting [Err4570].

See [RFC7700] for acknowledgements related to the specification that this document supersedes.

Author's Address

Peter Saint-Andre
Filament

Email: peter@filament.com
URI: https://filament.com/