Parameter Value Encoding in iCalendar and vCard

Abstract

   This specification updates the data formats for iCalendar (RFC 5545)
   and vCard (RFC 6350) to allow parameter values to include certain
   characters forbidden by the existing specifications.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6868.

Table of Contents

1. Introduction

   The iCalendar [RFC5545] specification defines a standard way to
   describe calendar data. The vCard [RFC6350] specification defines a
   standard way to describe contact data. Both of these use a similar
   text-based data format. Each iCalendar and vCard data object can
   include "properties" that have "parameters" and a "value". The value
   of a "parameter" is typically a token or URI value, but a "generic"
   text value is also allowed. However, the syntax rules for both
   iCalendar and vCard prevent the use of a double-quote character or
   control characters in such values, though double-quote characters and
   some subset of control characters are allowed in the actual property
   values.

   As more and more extensions are being developed for these data
   formats, there is a need to allow at least double-quotes and line
   feeds to be included in parameter values. The \-escaping mechanism
   used for property text values is not defined for use with parameter
   values and cannot be easily used in a backwards-compatible manner.
   This specification defines a new character escaping mechanism,
   compatible with existing parsers and chosen to minimize any impact on
   existing data.

2. Conventions Used in This Document

   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].

3. Parameter Value Encoding Scheme

   This specification defines the ^ character (U+005E -- Circumflex
   Accent) as an escape character in parameter values whose value type
   is defined using the "param-value" syntax element (Section 3.1 of
   iCalendar [RFC5545] and Section 3.3 of vCard [RFC6350]). The
   ^-escaping mechanism can be used when the value is either unquoted or
   quoted (i.e., whether or not the value is surrounded by double-
   quotes).

   When generating iCalendar or vCard parameter values, the following
   apply:

   o formatted text line breaks are encoded into ^n (U+005E, U+006E)

   o the ^ character (U+005E) is encoded into ^^ (U+005E, U+005E)

   o the " character (U+0022) is encoded into ^' (U+005E, U+0027)

   When parsing iCalendar or vCard parameter values, the following
   apply:

   o the character sequence ^n (U+005E, U+006E) is decoded into an
      appropriate formatted line break according to the type of system
      being used

   o the character sequence ^^ (U+005E, U+005E) is decoded into the ^
      character (U+005E)

   o the character sequence ^' (U+005E, U+0027) is decoded into the "
      character (U+0022)

   o if a ^ (U+005E) character is followed by any character other than
      the ones above, parsers MUST leave both the ^ and the following
      character in place

   When converting between iCalendar and vCard text-based data formats
   and alternative data-format representations such as XML (as described
   in [RFC6321] and [RFC6351], respectively), implementations MUST
   ensure that parameter value escape sequences are generated correctly
   in the text-based format and are decoded when the parameter values
   appear in the alternate data formats.

3.1. iCalendar Example

The following example is an "ATTENDEE" property with a "CN" parameter
whose value includes two double-quote characters. The parameter
value is not quoted, as there are no characters in the value that
would trigger quoting as required by iCalendar.

ATTENDEE;CN=George Herman ^'Babe^' Ruth:mailto:babe@example.com

The unescaped parameter value is

George Herman "Babe" Ruth

3.2. vCard Example

The following example is a "GEO" property with an "X-ADDRESS"
parameter whose value includes several line feed characters. The
parameter value is also quoted, since it contains a comma, which
triggers quoting as required by vCard.

GEO;X-ADDRESS="Pittsburgh Pirates^n115 Federal St^nPitt
 sburgh, PA 15212":geo:40.446816,-80.00566

The unescaped parameter value (where each line is terminated by a
line break character sequence) is

Pittsburgh Pirates
115 Federal St
Pittsburgh, PA 15212

4. Security Considerations

There are no additional security issues beyond those of iCalendar
[RFC5545] and vCard [RFC6350].

5. Acknowledgments

Thanks to Michael Angstadt, Tim Bray, Mike Douglass, Barry Leiba,
Simon Perreault, and Pete Resnick for feedback on this specification.

6. Normative References

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

   [RFC5545]   Desruisseaux, B., "Internet Calendaring and Scheduling
               Core Object Specification (iCalendar)", RFC 5545,
               September 2009.

   [RFC6321]   Daboo, C., Douglass, M., and S. Lees, "xCal: The XML
               Format for iCalendar", RFC 6321, August 2011.

   [RFC6350]   Perreault, S., "vCard Format Specification", RFC 6350,
               August 2011.

   [RFC6351]   Perreault, S., "xCard: vCard XML Representation",
               RFC 6351, August 2011.

Appendix A. Choice of Quoting Mechanism

   Having recognized the need for escaping parameter values, the
   question is what mechanism to use? One obvious choice would be to
   adopt the \-escaping used for property values. However, that could
   not be used as-is, because it escapes a double-quote as the sequence
   of \ followed by double-quote. Consider what the example in
   Section 3.1 might look like using \-escaping:

   ATTENDEE;CN="George Herman \"Babe\" Ruth":mailto:babe@example.com

   Existing iCalendar/vCard parsers know nothing about escape sequences
   in parameters. So they would parse the parameter value as:

   George Herman \

   i.e., the text between the first and second occurrence of a double-
   quote. However, the text after the second double-quote ought to be
   either a : or a ; (to delimit the parameter value from the following
   parameter or property) but is not, so the parser could legitimately
   throw an error at that point because the data is syntactically
   invalid. Thus, for backwards-compatibility reasons, a double-quote
   cannot be escaped using a sequence that itself includes a double-
   quote, and hence the choice of using a single-quote in this
   specification.

   Another option would be to use a form of \-escaping modified for use
   in parameter values only. However, some incorrect, non-interoperable
   use of \ in parameter values has been observed, and thus it is best
   to steer clear of that to achieve guaranteed, reliable
   interoperability. Also, given that double-quote gets changed to
   single-quote in the escape sequence for a parameter, but not for a
   value, it is better to not give the impression that the same escape
   mechanism (and thus code) can be used for both (which could lead to
   other issues, such as an implementation incorrectly escaping a ; as
   \; as opposed to quoting the parameter value).

   The choice of ^ as the escape character was made based on the
   requirement that an ASCII symbol (non-alphanumeric character) be
   used, and it ought to be one least likely to be found in existing
   data.

Author's Address

   Cyrus Daboo
   Apple Inc.
   1 Infinite Loop
   Cupertino, CA 95014
   USA

   EMail: cyrus@daboo.name
   URI: http://www.apple.com/