

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 12, 2021

P. Faltstrom
Netnod
F. Ljunggren
Kirei
D. van Gulik
Webweaving
March 11, 2021

The Base45 Data Encoding
draft-faltstrom-base45-01

Abstract

This document describes the base 45 encoding scheme which is built upon the base 64, base 32 and base 16 encoding schemes.

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 <https://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 September 12, 2021.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Conventions Used in This Document	2
3.	Interpretation of Encoded Data	2
4.	The Base 45 Encoding	2
4.1.	Encoding example	3
5.	IANA Considerations	3
6.	Security Considerations	3
7.	Acknowledgements	4
8.	Normative References	4
	Authors' Addresses	4

[1.](#) Introduction

When using QR or Aztec codes a different encoding scheme is needed than the already established base 64, base 32 and base 16 encoding schemes that are described in [RFC 4648](#) [[RFC4648](#)]. The difference from those and base 45 is the key table and that the padding with '=' is not required.

[2.](#) Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

[3.](#) Interpretation of Encoded Data

Encoded data is to be interpreted as described in [RFC 4648](#) [[RFC4648](#)] with the exception that a different alphabet is selected.

[4.](#) The Base 45 Encoding

A 45-character subset of US-ASCII is used, the 45 characters that can be used in a QR or Aztec code. If we look at Base 64, it encodes 3 bytes in 4 characters. Base 45 encodes 2 bytes over 3 characters.

The two bytes [A, B] are turned into [C, D, E] where $(A*256) + B = (C*45*45) + (D*45) + E$. The values C, D and E are then looked up in Table 1 to produce a three character string and the reverse when decoding.

If the number of octets are not dividable by two, the last remaining byte is represented by two characters.

Table 1: The Base 45 Alphabet

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
00	0	12	C	24	O	36	Space
01	1	13	D	25	P	37	\$
02	2	14	E	26	Q	38	%
03	3	15	F	27	R	39	*
04	4	16	G	28	S	40	+
05	5	17	H	29	T	41	-
06	6	18	I	30	U	42	.
07	7	19	J	31	V	43	/
08	8	20	K	32	W	44	:
09	9	21	L	33	X		
10	A	22	M	34	Y		
11	B	23	N	35	Z		

[4.1.](#) Encoding example

A series of bytes is turned into groups of two. Each such 16 bit value is turned into a series of three values calculated by doing successive calculations modulo 45. The values are in turned looked up in what is displayed in Table 1.

Example: The string "Hello!" is the byte sequence [72, 101, 108, 108, 111, 33]. If we look at each 16 bit value, it is [18633, 27756, 28449]. When looking at the values modulo 45, we get [[9, 9, 3], [13, 30, 36], [14, 2, 9]]. By looking up these values in the table we get the encoded string "993DU E29".

[5.](#) IANA Considerations

There are no considerations for IANA in this document.

[6.](#) Security Considerations

When implementing encoding and decoding it is important to be very careful so that buffer overflow does not take place, or anything similar. This includes of course the calculations of modulo 45 and lookup in the table of characters. Decoder also must be robust regarding input, including proper handling of the NUL character (ASCII 0).

Specifically it should be noted that Base 64 (for example) pad the string so that the encoding has the correct number of characters. This is something that Base 45 does not do, i.e. Base 45 do not include padding. Because of this, special care is to be taken when odd number of octets are to be encoded which results not in $N*3$

characters, but $(N-1)*3+2$ characters in the encoded string and vice versa, when the number of encoded characters are not divisible by 3.

7. Acknowledgements

The authors thank everyone that have been working with Base64 during the years that have proven the implementations are stable.

8. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", [RFC 4648](#), DOI 10.17487/RFC4648, October 2006, <<https://www.rfc-editor.org/info/rfc4648>>.

Authors' Addresses

Patrik Faltstrom
Netnod

Email: paf@netnod.se

Fredrik Ljunggren
Kirei

Email: fredrik@kirei.se

Dirk-Willem van Gulik
Webweaving

Email: dirkx@webweaving.org

