

Network Working Group
Internet-Draft
Expires: February 20, 2001

S.A. Josefsson (editor)
Extundo
August 22, 2000

Base 64, 32 and 16 Encodings
draft-josefsson-base-encoding-00.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>.

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

This Internet-Draft will expire on February 20, 2001.

Copyright Notice

Copyright (C) The Internet Society (2000). All Rights Reserved.

Abstract

This draft serve as a canonical collection of base 16, base 32 and base 64 encoding descriptions.

Internet-Draft

Base 64, 32 and 16 Encodings

August 2000

Table of Contents

1.	Base 64 Encoding	3
2.	Base 32 Encoding	5
3.	Base 16 Encoding	7
4.	Security Considerations	8
	References	9
	Author's Address	9
	Full Copyright Statement	10

[1.](#) Base 64 Encoding

The following description of base64 is due to [\[1\]](#) and [\[2\]](#).

A 65-character subset of US-ASCII is used, enabling 6 bits to be represented per printable character. (The extra 65th character, "=", is used to signify a special processing function.)

The encoding process represents 24-bit groups of input bits as output strings of 4 encoded characters. Proceeding from left to right, a 24-bit input group is formed by concatenating 3 8-bit input groups. These 24 bits are then treated as 4 concatenated 6-bit groups, each of which is translated into a single digit in the base 64 alphabet.

Each 6-bit group is used as an index into an array of 64 printable characters. The character referenced by the index is placed in the output string.

Table 1: The Base 64 Alphabet

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
0	A	17	R	34	i	51	z
1	B	18	S	35	j	52	0
2	C	19	T	36	k	53	1
3	D	20	U	37	l	54	2
4	E	21	V	38	m	55	3
5	F	22	W	39	n	56	4
6	G	23	X	40	o	57	5
7	H	24	Y	41	p	58	6
8	I	25	Z	42	q	59	7
9	J	26	a	43	r	60	8
10	K	27	b	44	s	61	9
11	L	28	c	45	t	62	+
12	M	29	d	46	u	63	/

13 N	30 e	47 v	
14 O	31 f	48 w	(pad) =
15 P	32 g	49 x	
16 Q	33 h	50 y	

Special processing is performed if fewer than 24 bits are available at the end of the data being encoded. A full encoding quantum is always completed at the end of a quantity. When fewer than 24 input bits are available in an input group, zero bits are added (on the right) to form an integral number of 6-bit groups. Padding at the end of the data is performed using the '=' character. Since all base 64 input is an integral number of octets, only the following cases can arise:

(1) the final quantum of encoding input is an integral multiple of 24 bits; here, the final unit of encoded output will be an integral multiple of 4 characters with no "=" padding,

(2) the final quantum of encoding input is exactly 8 bits; here, the final unit of encoded output will be two characters followed by two "=" padding characters, or

(3) the final quantum of encoding input is exactly 16 bits; here, the final unit of encoded output will be three characters followed by one "=" padding character.

Any characters outside of the base64 alphabet are to be ignored in base64-encoded data.

[2.](#) Base 32 Encoding

The following description of base32 is due to [\[3\]](#), the padding section was updated to fix two typos.

The Base32 encoding is designed to represent arbitrary sequences of octets in a form that needs to be case insensitive but need not be humanly readable.

A 33-character subset of US-ASCII is used, enabling 5 bits to be represented per printable character. (The extra 33rd character, "=", is used to signify a special processing function.)

The encoding process represents 40-bit groups of input bits as output strings of 8 encoded characters. Proceeding from left to right, a 40-bit input group is formed by concatenating 5 8bit input groups. These 40 bits are then treated as 8 concatenated 5-bit groups, each of which is translated into a single digit in the base32 alphabet. When encoding a bit stream via the base32 encoding, the bit stream must be presumed to be ordered with the most-significant-bit first. That is, the first bit in the stream

will be the high-order bit in the first 8bit byte, and the eighth bit will be the low-order bit in the first 8bit byte, and so on.

Each 5-bit group is used as an index into an array of 32 printable characters. The character referenced by the index is placed in the output string. These characters, identified in Table 2, below, are selected from US-ASCII digits and uppercase letters.

Table 2: The Base32 Alphabet

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
0	A	9	J	18	S	27	3
1	B	10	K	19	T	28	4
2	C	11	L	20	U	29	5
3	D	12	M	21	V	30	6
4	E	13	N	22	W	31	7
5	F	14	O	23	X		
6	G	15	P	24	Y	(pad)	=
7	H	16	Q	25	Z		
8	I	17	R	26	2		

Special processing is performed if fewer than 40 bits are available at the end of the data being encoded. A full encoding quantum is always completed at the end of a body. When fewer than 40 input bits are available in an input group, zero bits are added (on the right) to form an integral number of 5-bit groups. Padding at the end of the data is performed using the "=" character. Since all base32

input is an integral number of octets, only the following cases can arise:

- (1) the final quantum of encoding input is an integral multiple of 40 bits; here, the final unit of encoded output will be an integral multiple of 8 characters with no "=" padding,
- (2) the final quantum of encoding input is exactly 8 bits; here, the final unit of encoded output will be two characters followed by six "=" padding characters,
- (3) the final quantum of encoding input is exactly 16 bits; here, the final unit of encoded output will be four characters followed by four "=" padding characters,

(4) the final quantum of encoding input is exactly 24 bits; here, the final unit of encoded output will be five characters followed by three "=" padding characters, or

(5) the final quantum of encoding input is exactly 32 bits; here, the final unit of encoded output will be seven characters followed by one "=" padding character.

Because it is used only for padding at the end of the data, the occurrence of any "=" characters may be taken as evidence that the end of the data has been reached (without truncation in transit). No such assurance is possible, however, when the number of octets transmitted was a multiple of three and no "=" characters are present.

Any characters outside of the base32 alphabet are to be ignored in base32-encoded data.

[3.](#) Base 16 Encoding

The following description is original but analog to previous descriptions.

A 16-character subset of US-ASCII is used, enabling 4 bits to be represented per printable character.

The encoding process represents 8-bit groups (octets) of input bits as output strings of 2 encoded characters. Proceeding from left to right, a 8-bit input is taken from the input data. These 8 bits are then treated as 2 concatenated 4-bit groups, each of which is translated into a single digit in the base 16 alphabet.

Each 4-bit group is used as an index into an array of 16 printable characters. The character referenced by the index is placed in the output string.

Table 3: The Base 16 Alphabet

Value	Encoding	Value	Encoding	Value	Encoding	Value	Encoding
0	A	5	F	10	K	15	P
1	B	6	G	11	L	16	Q
2	C	7	H	12	M		
3	D	8	I	13	N		
4	E	9	J	14	O		

Unlike base32 and base64, no special padding is necessary since a full code words are always available.

Any characters outside of the base32 alphabet are to be ignored in base32-encoded data.

[4.](#) Security Considerations

This draft does not discuss security.

Internet-Draft

Base 64, 32 and 16 Encodings

August 2000

References

- [1] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", November 1996.
- [2] Eastlake, D., "Domain Name System Security Extensions", March 1999.
- [3] Myers, J., "SASL GSSAPI mechanisms", May 2000.

Author's Address

Simon Josefsson
Extundo
Drottningholmsvägen 70
Stockholm 112 42
Sweden

Phone: +46 8 6190422
EMail: simon@josefsson.org

Internet-Draft

Base 64, 32 and 16 Encodings

August 2000

Full Copyright Statement

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Acknowledgement

Funding for the RFC editor function is currently provided by the Internet Society.

Josefsson (editor)

Expires February 20, 2001

[Page 10]