Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'setvbuf.3p' command

$ man setvbuf.3p

SETVBUF(3P) POSIX Programmer's Manual SETVBUF(3P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

setvbuf ? assign buffering to a stream

SYNOPSIS

```
#include <stdio.h>
int setvbuf(FILE *restrict stream, char *restrict buf, int type,
    size_t size);
```

DESCRIPTION

The functionality described on this reference page is aligned with the ISO C standard. Any conflict between the requirements described here and the ISO C standard is unintentional. This volume of POSIX.1?2017 defers to the ISO C standard.

The setvbuf() function may be used after the stream pointed to by stream is associated with an open file but before any other operation (other than an unsuccessful call to setvbuf()) is performed on the stream. The argument type determines how stream shall be buffered, as follows:

* {_IOFBF} shall cause input/output to be fully buffered.

* {_IOLBF} shall cause input/output to be line buffered.

* {_IONBF} shall cause input/output to be unbuffered.

If buf is not a null pointer, the array it points to may be used in? stead of a buffer allocated by setvbuf() and the argument size speci? fies the size of the array; otherwise, size may determine the size of a buffer allocated by the setvbuf() function. The contents of the array at any time are unspecified.

For information about streams, see Section 2.5, Standard I/O Streams.

RETURN VALUE

Upon successful completion, setvbuf() shall return 0. Otherwise, it shall return a non-zero value if an invalid value is given for type or if the request cannot be honored, and may set errno to indicate the er? ror.

ERRORS

The setvbuf() function may fail if:

EBADF The file descriptor underlying stream is not valid.

The following sections are informative.

EXAMPLES

None.

APPLICATION USAGE

A common source of error is allocating buffer space as an ``automatic'' variable in a code block, and then failing to close the stream in the same block.

With setvbuf(), allocating a buffer of size bytes does not necessarily imply that all of size bytes are used for the buffer area.

Applications should note that many implementations only provide line buffering on input from terminal devices.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.5, Standard I/O Streams, fopen(), setbuf()

The Base Definitions volume of POSIX.1?2017, <stdio.h>

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Por? table Operating System Interface (POSIX), The Open Group Base Specifi? cations Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at http://www.opengroup.org/unix/online.html .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.ker? nel.org/doc/man-pages/reporting_bugs.html .