



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

\$ man writev.3p

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

writev ? write a vector

SYNOPSIS

```
#include <sys/uio.h>
```

```
ssize_t writev(int fildes, const struct iovec *iov, int iovcnt);
```

DESCRIPTION

The `writev()` function shall be equivalent to `write()`, except as described below. The `writev()` function shall gather output data from the `iovcnt` buffers specified by the members of the `iov` array: `iov[0]`, `iov[1]`, ..., `iov[iovcnt-1]`. The `iovcnt` argument is valid if greater than 0 and less than or equal to `{IOV_MAX}`, as defined in `<limits.h>`. Each `iovec` entry specifies the base address and length of an area in memory from which data should be written. The `writev()` function shall always write a complete area before proceeding to the next.

If `fildes` refers to a regular file and all of the `iov_len` members in the array pointed to by `iov` are 0, `writev()` shall return 0 and have no other effect. For other file types, the behavior is unspecified.

If the sum of the `iov_len` values is greater than `{SSIZE_MAX}`, the operation shall fail and no data shall be transferred.

RETURN VALUE

Upon successful completion, `writev()` shall return the number of bytes actually written. Otherwise, it shall return a value of `-1`, the file-pointer shall remain unchanged, and `errno` shall be set to indicate an error.

ERRORS

Refer to `write()`.

In addition, the `writev()` function shall fail if:

EINVAL The sum of the `iov_len` values in the `iov` array would overflow an `ssize_t`.

The `writev()` function may fail and set `errno` to:

EINVAL The `iovcnt` argument was less than or equal to 0, or greater than `{IOV_MAX}`.

The following sections are informative.

EXAMPLES

Writing Data from an Array

The following example writes data from the buffers specified by members of the `iov` array to the file associated with the file descriptor `fd`.

```
#include <sys/types.h>
#include <sys/uio.h>
#include <unistd.h>
...
ssize_t bytes_written;
int fd;
char *buf0 = "short string\n";
char *buf1 = "This is a longer string\n";
char *buf2 = "This is the longest string in this example\n";
int iovcnt;
struct iovec iov[3];
iov[0].iov_base = buf0;
iov[0].iov_len = strlen(buf0);
```

```
iov[1].iov_base = buf1;
iov[1].iov_len = strlen(buf1);
iov[2].iov_base = buf2;
iov[2].iov_len = strlen(buf2);
...
iocnt = sizeof(iov) / sizeof(struct iovec);
bytes_written = writev(fd, iov, iocnt);
...
```

APPLICATION USAGE

None.

RATIONALE

Refer to write().

FUTURE DIRECTIONS

None.

SEE ALSO

readv(), write()

The Base Definitions volume of POSIX.1-2017, <limits.h>, <sys_uio.h>

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications 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.kernel.org/doc/man-pages/reporting_bugs.html.