



*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 'mq\_notify.3p' command***

***\$ man mq\_notify.3p***

MQ\_NOTIFY(3P)      POSIX Programmer's Manual      MQ\_NOTIFY(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

mq\_notify ? notify process that a message is available (REALTIME)

### SYNOPSIS

```
#include <mqueue.h>

int mq_notify(mqd_t mqdes, const struct sigevent *notification);
```

### DESCRIPTION

If the argument notification is not NULL, this function shall register the calling process to be notified of message arrival at an empty message queue associated with the specified message queue descriptor, mqdes. The notification specified by the notification argument shall be sent to the process when the message queue transitions from empty to non-empty. At any time, only one process may be registered for notification by a message queue. If the calling process or any other process has already registered for notification of message arrival at the specified message queue, subsequent attempts to register for that message queue shall fail.

If notification is NULL and the process is currently registered for no?

tification by the specified message queue, the existing registration shall be removed.

When the notification is sent to the registered process, its registration shall be removed. The message queue shall then be available for registration.

If a process has registered for notification of message arrival at a message queue and some thread is blocked in `mq_receive()` or `mq_timedreceive()` waiting to receive a message when a message arrives at the queue, the arriving message shall satisfy the appropriate `mq_receive()` or `mq_timedreceive()`, respectively. The resulting behavior is as if the message queue remains empty, and no notification shall be sent.

## RETURN VALUE

Upon successful completion, the `mq_notify()` function shall return a value of zero; otherwise, the function shall return a value of -1 and set `errno` to indicate the error.

## ERRORS

The `mq_notify()` function shall fail if:

**EBADF** The `mqdes` argument is not a valid message queue descriptor.

**EBUSY** A process is already registered for notification by the message queue.

The `mq_notify()` function may fail if:

**EINVAL** The `notification` argument is `NULL` and the process is currently not registered.

The following sections are informative.

## EXAMPLES

The following program registers a notification request for the message queue named in its command-line argument. Notification is performed by creating a thread. The thread executes a function which reads one message from the queue and then terminates the process.

```
#include <pthread.h>
#include <mqueue.h>
#include <assert.h>
#include <stdio.h>
```

```

#include <stdlib.h>

#include <unistd.h>

static void          /* Thread start function */
tfunc(union sigval sv)
{
    struct mq_attr attr;
    ssize_t nr;
    void *buf;

    mqd_t mqdes = *((mqd_t *) sv.sival_ptr);

    /* Determine maximum msg size; allocate buffer to receive msg */
    if (mq_getattr(mqdes, &attr) == -1) {
        perror("mq_getattr");
        exit(EXIT_FAILURE);
    }

    buf = malloc(attr.mq_msgsize);

    if (buf == NULL) {
        perror("malloc");
        exit(EXIT_FAILURE);
    }

    nr = mq_receive(mqdes, buf, attr.mq_msgsize, NULL);

    if (nr == -1) {
        perror("mq_receive");
        exit(EXIT_FAILURE);
    }

    printf("Read %ld bytes from message queue\n", (long) nr);

    free(buf);

    exit(EXIT_SUCCESS);    /* Terminate the process */
}

int
main(int argc, char *argv[])
{
    mqd_t mqdes;
    struct sigevent not;

```



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](https://www.kernel.org/doc/man-pages/reporting_bugs.html) .

IEEE/The Open Group

2017

MQ\_NOTIFY(3P)