



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 'getpwnam.3p' command

\$ man getpwnam.3p

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

getpwnam, getpwnam_r ? search user database for a name

SYNOPSIS

```
#include <pwd.h>

struct passwd *getpwnam(const char *name);

int getpwnam_r(const char *name, struct passwd *pwd, char *buffer,
               size_t bufsize, struct passwd **result);
```

DESCRIPTION

The `getpwnam()` function shall search the user database for an entry with a matching name.

The `getpwnam()` function need not be thread-safe.

Applications wishing to check for error situations should set `errno` to 0 before calling `getpwnam()`. If `getpwnam()` returns a null pointer and `errno` is non-zero, an error occurred.

The `getpwnam_r()` function shall update the `passwd` structure pointed to by `pwd` and store a pointer to that structure at the location pointed to by `result`. The structure shall contain an entry from the user database

with a matching name. Storage referenced by the structure is allocated from the memory provided with the buffer parameter, which is bufsize bytes in size. A call to `sysconf(_SC_GETPW_R_SIZE_MAX)` returns either -1 without changing `errno` or an initial value suggested for the size of this buffer. A null pointer shall be returned at the location pointed to by result on error or if the requested entry is not found.

RETURN VALUE

The `getpwnam()` function shall return a pointer to a struct `passwd` with the structure as defined in `<pwd.h>` with a matching entry if found. A null pointer shall be returned if the requested entry is not found, or an error occurs. If the requested entry was not found, `errno` shall not be changed. On error, `errno` shall be set to indicate the error.

The application shall not modify the structure to which the return value points, nor any storage areas pointed to by pointers within the structure. The returned pointer, and pointers within the structure, might be invalidated or the structure or the storage areas might be overwritten by a subsequent call to `getpwent()`, `getpwnam()`, or `getpwuid()`. The returned pointer, and pointers within the structure, might also be invalidated if the calling thread is terminated.

The `getpwnam_r()` function shall return zero on success or if the requested entry was not found and no error has occurred. If an error has occurred, an error number shall be returned to indicate the error.

ERRORS

These functions may fail if:

EIO An I/O error has occurred.

EINTR A signal was caught during `getpwnam()`.

EMFILE All file descriptors available to the process are currently open.

ENFILE The maximum allowable number of files is currently open in the system.

The `getpwnam_r()` function may fail if:

ERANGE Insufficient storage was supplied via buffer and bufsize to contain the data to be referenced by the resulting `passwd` struct?

ture.

The following sections are informative.

EXAMPLES

Note that `sysconf(_SC_GETPW_R_SIZE_MAX)` may return -1 if there is no hard limit on the size of the buffer needed to store all the groups returned. This example shows how an application can allocate a buffer of sufficient size to work with `getpwnam_r()`.

```
long int initlen = sysconf(_SC_GETPW_R_SIZE_MAX);
size_t len;
if (initlen == -1)
    /* Default initial length. */
    len = 1024;
else
    len = (size_t) initlen;
struct passwd result;
struct passwd *resultp;
char *buffer = malloc(len);
if (buffer == NULL)
    ...handle error...
int e;
while ((e = getpwnam_r("someuser", &result, buffer, len, &resultp))
    == ERANGE)
{
    size_t newlen = 2 * len;
    if (newlen < len)
        ...handle error...
    len = newlen;
    char *newbuffer = realloc(buffer, len);
    if (newbuffer == NULL)
        ...handle error...
    buffer = newbuffer;
}
if (e != 0)
```

...handle error...

```
free (buffer);
```

Getting an Entry for the Login Name

The following example uses the `getlogin()` function to return the name of the user who logged in; this information is passed to the `getpwnam()` function to get the user database entry for that user.

```
#include <sys/types.h>
#include <pwd.h>
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
...
char *lgn;
struct passwd *pw;
...
if ((lgn = getlogin()) == NULL || (pw = getpwnam(lgn)) == NULL) {
    fprintf(stderr, "Get of user information failed.\n"); exit(1);
}
...
```

APPLICATION USAGE

Three names associated with the current process can be determined: `getpwnam(geteuid())` returns the name associated with the effective user ID of the process; `getlogin()` returns the name associated with the current login activity; and `getpwuid(getuid())` returns the name associated with the real user ID of the process.

The `getpwnam_r()` function is thread-safe and returns values in a user-supplied buffer instead of possibly using a static data area that may be overwritten by each call.

Portable applications should take into account that it is usual for an implementation to return -1 from `sysconf()` indicating that there is no maximum for `_SC_GETPW_R_SIZE_MAX`.

RATIONALE

None.

FUTURE DIRECTIONS

None.

SEE ALSO

`getpwuid()`, `sysconf()`

The Base Definitions volume of POSIX.1?2017, `<pwd.h>`, `<sys_types.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 .

IEEE/The Open Group

2017

GETPWNAM(3P)