



**Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!**

***Rocky Enterprise Linux 9.2 Manual Pages on command 'getnetbyaddr\_r.3'***

**\$ man getnetbyaddr\_r.3**

GETNETENT\_R(3)

Linux Programmer's Manual

GETNETENT\_R(3)

**NAME**

getnetent\_r, getnetbyname\_r, getnetbyaddr\_r - get network entry (reentrant)

**SYNOPSIS**

```
#include <netdb.h>

int getnetent_r(struct netent *result_buf, char *buf,
                 size_t buflen, struct netent **result,
                 int *h_errno);

int getnetbyname_r(const char *name,
                   struct netent *result_buf, char *buf,
                   size_t buflen, struct netent **result,
                   int *h_errno);

int getnetbyaddr_r(uint32_t net, int type,
                   struct netent *result_buf, char *buf,
                   size_t buflen, struct netent **result,
                   int *h_errno);
```

Feature Test Macro Requirements for glibc (see `feature_test_macros(7)`):

`getnetent_r(), getnetbyname_r(), getnetbyaddr_r():`

Since glibc 2.19:

`_DEFAULT_SOURCE`

Glibc 2.19 and earlier:

`_BSD_SOURCE || _SVID_SOURCE`

The `getnetent_r()`, `getnetbyname_r()`, and `getnetbyaddr_r()` functions are the reentrant equivalents of, respectively, `getnetent(3)`, `getnetbyname(3)`, and `getnetbynumber(3)`. They differ in the way that the netent structure is returned, and in the function calling `sig?` nature and return value. This manual page describes just the differences from the non-reentrant functions.

Instead of returning a pointer to a statically allocated netent structure as the function result, these functions copy the structure into the location pointed to by `result_buf`.

The `buf` array is used to store the string fields pointed to by the returned `netent` structure. (The nonreentrant functions allocate these strings in static storage.) The size of this array is specified in `buflen`. If `buf` is too small, the call fails with the error `ERANGE`, and the caller must try again with a larger buffer. (A buffer of length 1024 bytes should be sufficient for most applications.)

If the function call successfully obtains a network record, then `*result` is set pointing to `result_buf`; otherwise, `*result` is set to `NULL`.

The buffer pointed to by `h_errnop` is used to return the value that would be stored in the global variable `h_errno` by the nonreentrant versions of these functions.

## RETURN VALUE

On success, these functions return 0. On error, they return one of the positive error numbers listed in ERRORS.

On error, record not found (getnetbyname\_r(), getnetbyaddr\_r()), or end of input (getnetent\_r()) result is set to NULL.

## ERRORS

ENOENT (getnetent\_r()) No more records in database.

ERANGE buf is too small. Try again with a larger buffer (and increased buflen).

## ATTRIBUTES

For an explanation of the terms used in this section, see attributes(7).

3 Interface      3 Attribute      3 Value      3

3getnetpent\_r() 3 Thread safety 3 MT Safe locals 3

3-acetylbenzylamine (4) 3 3

3getnetbyaddr\_r() 3 3 3

## CONFORMING TO

These functions are GNU extensions. Functions with similar names exist on some other systems, though typically with different calling signatures.

## SEE ALSO

`getnetent(3)`, `networks(5)`

## COLOPHON

This page is part of release 5.10 of the Linux man-pages project. A description of the project, information about reporting bugs, and the latest version of this page, can be found at <https://www.kernel.org/doc/man-pages/>.

GNU

2017-09-15

GETNETENT\_R(3)