

NAME

getrpcnt, getrpcbyname, getrpcbnumber, setrpcnt, endrpcnt – get RPC entry

LIBRARY

Standard C library (*libc*, *-lc*)

SYNOPSIS

```
#include <netdb.h>

struct rpcnt *getrpcnt(void);

struct rpcnt *getrpcbyname(const char *name);
struct rpcnt *getrpcbnumber(int number);

void setrpcnt(int stayopen);
void endrpcnt(void);
```

DESCRIPTION

The **getrpcnt()**, **getrpcbyname()**, and **getrpcbnumber()** functions each return a pointer to an object with the following structure containing the broken-out fields of an entry in the RPC program number data base.

```
struct rpcnt {
    char *r_name;      /* name of server for this RPC program */
    char **r_aliases;  /* alias list */
    long r_number;     /* RPC program number */
};
```

The members of this structure are:

r_name The name of the server for this RPC program.

r_aliases

A NULL-terminated list of alternate names for the RPC program.

r_number

The RPC program number for this service.

The **getrpcnt()** function reads the next entry from the database. A connection is opened to the database if necessary.

The **setrpcnt()** function opens a connection to the database, and sets the next entry to the first entry. If *stayopen* is nonzero, then the connection to the database will not be closed between calls to one of the **getrpc***() functions.

The **endrpcnt()** function closes the connection to the database.

The **getrpcbyname()** and **getrpcbnumber()** functions sequentially search from the beginning of the file until a matching RPC program name or program number is found, or until end-of-file is encountered.

RETURN VALUE

On success, **getrpcnt()**, **getrpcbyname()**, and **getrpcbnumber()** return a pointer to a statically allocated *rpcnt* structure. NULL is returned on EOF or error.

FILES

/etc/rpc

RPC program number database.

ATTRIBUTES

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

Interface	Attribute	Value
getrpcnt() , getrpcbyname() , getrpcbnumber()	Thread safety	MT-Unsafe
setrpcnt() , endrpcnt()	Thread safety	MT-Safe locale

STANDARDS

Not in POSIX.1. Present on the BSDs, Solaris, and many other systems.

BUGS

All information is contained in a static area so it must be copied if it is to be saved.

SEE ALSO

getrpcnt_r(3), **rpc(5)**, **rpcinfo(8)**, **ypserv(8)**