

NAME

getprotoent, **getprotobynumber**, **getprotobyname**, **setprotoent**, **endprotoent** - get protocol entry

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <netdb.h>
```

```
struct protoent *
```

```
getprotoent(void);
```

```
struct protoent *
```

```
getprotobyname(const char *name);
```

```
struct protoent *
```

```
getprotobynumber(int proto);
```

```
void
```

```
setprotoent(int stayopen);
```

```
void
```

```
endprotoent(void);
```

DESCRIPTION

The **getprotoent**(), **getprotobyname**(), and **getprotobynumber**() functions each return a pointer to an object with the following structure containing the broken-out fields of a line in the network protocol data base, */etc/protocols*.

```
struct protoent {
    char    *p_name; /* official name of protocol */
    char    **p_aliases; /* alias list */
    int     p_proto; /* protocol number */
};
```

The members of this structure are:

p_name The official name of the protocol.

p_aliases A zero terminated list of alternate names for the protocol.

p_proto The protocol number.

The **getprotoent()** function reads the next line of the file, opening the file if necessary.

The **setprotoent()** function opens and rewinds the file. If the *stayopen* flag is non-zero, the net data base will not be closed after each call to **getprotobyname()** or **getprotobynumber()**.

The **endprotoent()** function closes the file.

The **getprotobyname()** function and **getprotobynumber()** sequentially search from the beginning of the file until a matching protocol name or protocol number is found, or until EOF is encountered.

RETURN VALUES

Null pointer returned on EOF or error.

FILES

/etc/protocols

SEE ALSO

protocols(5)

HISTORY

The **getprotoent()**, **getprotobynumber()**, **getprotobyname()**, **setprotoent()**, and **endprotoent()** functions appeared in 4.2BSD.

BUGS

These functions use a thread-specific data space; if the data is needed for future use, it should be copied before any subsequent calls overwrite it. Only the Internet protocols are currently understood.