

**NAME**

getservent, getservbyname, getservbyport, setservent, endservent – get service entry

**LIBRARY**

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

**SYNOPSIS**

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

```
struct servent *getservent(void);
```

```
struct servent *getservbyname(const char *name, const char *proto);
```

```
struct servent *getservbyport(int port, const char *proto);
```

```
void setservent(int stayopen);
```

```
void endservent(void);
```

**DESCRIPTION**

The **getservent()** function reads the next entry from the services database (see **services(5)**) and returns a *servent* structure containing the broken-out fields from the entry. A connection is opened to the database if necessary.

The **getservbyname()** function returns a *servent* structure for the entry from the database that matches the service *name* using protocol *proto*. If *proto* is NULL, any protocol will be matched. A connection is opened to the database if necessary.

The **getservbyport()** function returns a *servent* structure for the entry from the database that matches the port *port* (given in network byte order) using protocol *proto*. If *proto* is NULL, any protocol will be matched. A connection is opened to the database if necessary.

The **setservent()** 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 **get-serv\*()** functions.

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

The *servent* structure is defined in *<netdb.h>* as follows:

```
struct servent {
    char *s_name;           /* official service name */
    char **s_aliases;       /* alias list */
    int s_port;             /* port number */
    char *s_proto;          /* protocol to use */
}
```

The members of the *servent* structure are:

*s\_name*

The official name of the service.

*s\_aliases*

A NULL-terminated list of alternative names for the service.

*s\_port*

The port number for the service given in network byte order.

*s\_proto*

The name of the protocol to use with this service.

**RETURN VALUE**

The **getservent()**, **getservbyname()**, and **getservbyport()** functions return a pointer to a statically allocated *servent* structure, or NULL if an error occurs or the end of the file is reached.

**FILES**

*/etc/services*

services database file

## ATTRIBUTES

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

Interface	Attribute	Value
<b>getservent()</b>	Thread safety	MT-Unsafe race:servent race:serventbuf locale
<b>getservbyname()</b>	Thread safety	MT-Unsafe race:servbyname locale
<b>getservbyport()</b>	Thread safety	MT-Unsafe race:servbyport locale
<b>setservent(), endservent()</b>	Thread safety	MT-Unsafe race:servent locale

In the above table, *servent* in *race:servent* signifies that if any of the functions **setservent()**, **getservent()**, or **endservent()** are used in parallel in different threads of a program, then data races could occur.

## STANDARDS

POSIX.1-2001, POSIX.1-2008, 4.3BSD.

## SEE ALSO

**getnetent(3)**, **getprotoent(3)**, **getservent\_r(3)**, **services(5)**