

**NAME**

getusershell, setusershell, endusershell – get permitted user shells

**LIBRARY**

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

**SYNOPSIS**

```
#include <unistd.h>
```

```
char *getusershell(void);  
void setusershell(void);  
void endusershell(void);
```

Feature Test Macro Requirements for glibc (see **feature\_test\_macros(7)**):

**getusershell(), setusershell(), endusershell():**

Since glibc 2.21:

    \_DEFAULT\_SOURCE

In glibc 2.19 and 2.20:

    \_DEFAULT\_SOURCE || (\_XOPEN\_SOURCE && \_XOPEN\_SOURCE < 500)

Up to and including glibc 2.19:

    \_BSD\_SOURCE || (\_XOPEN\_SOURCE && \_XOPEN\_SOURCE < 500)

**DESCRIPTION**

The **getusershell()** function returns the next line from the file */etc/shells*, opening the file if necessary. The line should contain the pathname of a valid user shell. If */etc/shells* does not exist or is unreadable, **getusershell()** behaves as if */bin/sh* and */bin/csh* were listed in the file.

The **setusershell()** function rewinds */etc/shells*.

The **endusershell()** function closes */etc/shells*.

**RETURN VALUE**

The **getusershell()** function returns NULL on end-of-file.

**FILES**

*/etc/shells*

**ATTRIBUTES**

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

Interface	Attribute	Value
<b>getusershell(), setusershell(), endusershell()</b>	Thread safety	MT-Unsafe

**STANDARDS**

4.3BSD.

**SEE ALSO**

**shells(5)**