

NAME

getttyent, **getttynam**, **setttyent**, **endttyent**, **isdialuptty**, **isnettty** - ttys(5) file routines

LIBRARY

Standard C Library (libc, -lc)

SYNOPSIS

```
#include <ttyent.h>
```

```
struct ttyent *
```

```
getttyent(void);
```

```
struct ttyent *
```

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

```
int
```

```
setttyent(void);
```

```
int
```

```
endttyent(void);
```

```
int
```

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

```
int
```

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

DESCRIPTION

The **getttyent()**, and **getttynam()** functions each return a pointer to an object, with the following structure, containing the broken-out fields of a line from the tty description file.

```
struct ttyent {
    char    *ty_name;        /* terminal device name */
    char    *ty_getty;      /* command to execute, usually getty */
    char    *ty_type; /* terminal type for termcap */
#define TTY_ON        0x01    /* enable logins (start ty_getty program) */
#define TTY_SECURE    0x02    /* allow uid of 0 to login */
#define TTY_DIALUP    0x04    /* is a dialup tty */
#define TTY_NETWORK   0x08    /* is a network tty */
#define TTY_IFEXISTS  0x10    /* configured as "onifexists" */
```

```

#define TTY_IFCONSOLE0x20    /* configured as "onifconsole" */
int      ty_status; /* status flags */
char     *ty_window;      /* command to start up window manager */
char     *ty_comment;     /* comment field */
char     *ty_group;       /* tty group name */
};

```

The fields are as follows:

ty_name The name of the character-special file.

ty_getty The name of the command invoked by `init(8)` to initialize tty line characteristics.

ty_type The name of the default terminal type connected to this tty line.

ty_status A mask of bit fields which indicate various actions allowed on this tty line. The possible flags are as follows:

TTY_ON Enables logins (i.e., `init(8)` will start the command referenced by *ty_getty* on this entry).

TTY_SECURE Allow users with a uid of 0 to login on this terminal.

TTY_DIALUP Identifies a tty as a dialin line. If this flag is set, then **isdialuptty()** will return a non-zero value.

TTY_NETWORK Identifies a tty used for network connections. If this flag is set, then **isnettty()** will return a non-zero value.

TTY_IFEXISTS Identifies a tty that does not necessarily exist.

TTY_IFCONSOLE Identifies a tty that might be a system console.

ty_window The command to execute for a window system associated with the line.

ty_group A group name to which the tty belongs. If no group is specified in the ttys description file, then the tty is placed in an anonymous group called "none".

ty_comment Any trailing comment field, with any leading hash marks (“#”) or whitespace removed.

If any of the fields pointing to character strings are unspecified, they are returned as null pointers. The field *ty_status* will be zero if no flag values are specified.

See [ttys\(5\)](#) for a more complete discussion of the meaning and usage of the fields.

The **getttyent()** function reads the next line from the *ttys* file, opening the file if necessary. The **settttyent()** function rewinds the file if open, or opens the file if it is unopened. The **endtttyent()** function closes any open files.

The **getttynam()** function searches from the beginning of the file until a matching *name* is found (or until EOF is encountered).

RETURN VALUES

The routines **getttyent()** and **getttynam()** return a null pointer on EOF or error. The **settttyent()** function and **endtttyent()** return 0 on failure and 1 on success.

The routines **isdialuptty()** and **isnettty()** return non-zero if the dialup or network flag is set for the *tty* entry relating to the *tty* named by the argument, and zero otherwise.

FILES

/etc/ttys

SEE ALSO

[login\(1\)](#), [gettytab\(5\)](#), [termcap\(5\)](#), [ttys\(5\)](#), [getty\(8\)](#), [init\(8\)](#)

HISTORY

The **getttyent()**, **getttynam()**, **settttyent()**, and **endtttyent()** functions appeared in 4.3BSD.

BUGS

These functions use static data storage; if the data is needed for future use, it should be copied before any subsequent calls overwrite it.