

NAME

`getty` - set terminal type, modes, speed, and line discipline

SYNOPSIS

```
/etc/getty [ -h ] line [ speed [ type [ linedisc ] ] ]
/etc/getty -t gettydefs-like-file
```

DESCRIPTION

Getty is a program that is invoked by *init*(1). It is the second process in the series, *init-getty-login-shell*, that ultimately connects a terminal user with UNIX. Initially *getty* generates a system identification message from the values returned by the *uname*(2) system call. Then, if */etc/issue* exists, it outputs this to the user terminal, followed finally by the login message field for the entry it is using from */etc/gettydefs*. *Getty* reads the user's login name and invokes the *login*(1) command with the user's name as argument. While reading the name, *getty* attempts to adapt the system to the speed and type of terminal being used.

Line is the name of a tty line in */dev* to which *getty* is to attach itself. *Getty* uses this string as the name of a file in the */dev* directory to open for reading and writing. Unless *getty* is invoked with the *-h* flag, *getty* will force a hangup on the line by setting the speed to zero before setting the speed to the default or specified speed. The optional second argument, *speed*, is a label to a speed and tty definition in the file */etc/gettydefs*(5). This definition tells *getty* what speed to initially run at, what the login message should look like, what the initial tty settings are, and what speed to try next should the user indicate that the speed is inappropriate. (By typing a *<break>* character.) The default speed is 300 baud. The optional third argument, *type*, is a character string describing to *getty* what type of terminal is connected to the line in question. *Getty* understands the following types:

none	default
tec	TEC scope
vt61	DEC vt61
vt100	DEC vt100
tektonix	Tektronix
tek	Tektronix
ds40-1	Teletype DS40-1
hp45	Hewlett-Packard HP45
ds40-2b	Teletype DS40-2b

The default terminal is "**none**"; i.e., any crt or normal terminal unknown to the system. The optional fourth argument, *linedisc*, is a character string describing which line discipline to use in communicating with the terminal. *getty* understands the following line disciplines:

full_duplex	default
full	default
transparent	transparent (see <i>ioctl</i> (2))
trans	transparent
half_duplex	half duplex
half	half duplex
votrax	votrax

The default is "**full_duplex**"; i.e., line discipline zero.

When given no optional arguments, *getty* sets the speed of the interface to 300 baud, specifies that raw mode is to be used (awaken on every character), that echo is to be suppressed, either parity allowed, newline characters will be converted to carriage return-line feed, and tab expansion performed on the standard output. It types the login message before reading the user's name a character at a time. If a null character (or framing error) is received, it is assumed to be the result of the user pushing the "break" key. This will cause *getty* to attempt the next

speed in the series. The series that *getty* tries is determined by what it finds in */etc/gettydefs*.

The user's name is terminated by a new-line or carriage-return character. The latter results in the system being set to treat carriage returns appropriately (see *ioctl(2)*).

The user's name is scanned to see if it contains any lower-case alphabetic characters; if not, and if the name is non-empty, the system is told to map any future upper-case characters into the corresponding lower-case characters.

In addition to the standard UNIX erase and kill characters, '#' and '@', *getty* also understands '\b'. If the user uses a '\b' as a rubout, *getty* sets the standard erase character to backspace and the standard kill character to '@' instead of '#' and '@'.

Getty also understands the "standard" ESS protocols for erasing, killing and aborting a line, and terminating a line. If *getty* sees the ESS erase character, '_', or kill character, '\$', or abort character, '&', or the ESS line terminators, '/' or '!', it attempts to set up the terminal into STDTTY mode (see *ioctl(2)*), which has those characters as the erase, kill, and line terminator characters. If it doesn't succeed, the standard erase and kill characters will be used.

Finally, *login* is called with the user's name as an argument. Additional arguments may be typed after the login name. These are passed to *login*, which will place them in the environment. (see *login(1)*)

A test option is provided. When *getty* is invoked with the *-t* switch and a file, it scans the file as if it were scanning */etc/gettydefs* and prints out the results to the standard output. If there are any unrecognized modes or improperly constructed entries, it reports these. If the entries are correct, it prints out the values of the various flags. See *ioctl(2)* to interpret the values. Note that some values are added to the flags automatically.

FILES

/etc/gettydefs, */etc/issue*

SEE ALSO

ct(1), *init(1M)*, *login(1)*, *ioctl(2)*, *tty(4)*, *gettydefs(5)*, *inittab(5)*

BUGS

While *getty* does understand simple single character quoting conventions, it is not possible to quote the special control characters that *getty* uses to determine when the end of the line has been reached, which protocol is being used, and what the erase character is. Therefore it is not possible to login via *getty* and type a '#', '@', '/', '!', '_', backspace, '^D', or '&' in your response.

