

**NAME**

nohup - run a command immune to hangups

**SYNOPSIS**

**nohup** *command* [*arguments*]

**DESCRIPTION**

**nohup** executes *command* with hangups and quits ignored. If output is not redirected by the user, both standard output and standard error are sent to **nohup.out**. If **nohup.out** is not writable in the current directory, output is redirected to **\$HOME/nohup.out**; otherwise, **nohup** fails. If a file is created, the file's permission bits will be set to **S\_IRUSR** | **S\_IWUSR**.

If output from **nohup** is redirected to a terminal, or is not redirected at all, the output is sent to **nohup.out**.

**EXTERNAL INFLUENCES****Environment Variables**

**LC\_MESSAGES** determines the language in which messages are displayed.

If **LC\_MESSAGES** is not specified in the environment or is set to the empty string, the value of **LANG** is used as a default for each unspecified or empty variable. If **LANG** is not specified or is set to the empty string, a default of "C" (see *lang(5)*) is used instead of **LANG**.

If any internationalization variable contains an invalid setting, **nohup** behaves as if all internationalization variables are set to "C". See *environ(5)*.

**International Code Set Support**

Single- and multi-byte character code sets are supported.

**EXAMPLES**

It is frequently desirable to apply **nohup** to pipelines or lists of commands. This can be done only by placing pipelines and command lists in a single file, called a shell script. To run the script using **nohup**:

```
nohup sh file
```

**nohup** features apply to the entire contents of *file*. If the shell script *file* is to be executed often, the need to type **sh** can be eliminated by setting execute permission on *file*. The script can also be run in the background with interrupts ignored (see *sh(1)*):

```
nohup file &
```

*file* typically contains normal keyboard command sequences that one would want to continue running in case the terminal disconnects, such as:

```
tbl ofile | eqn | nroff > nfile
```

**WARNINGS**

Be careful to place punctuation properly. For example, in the command form:

```
nohup command1; command2
```

**nohup** applies only to *command1*. To correct the problem, use the command form:

```
nohup (command1; command2)
```

Be careful of where standard error is redirected. The following command may put error messages on tape, making it unreadable:

```
nohup cpio -o <list >/dev/rmt/c0t0d0BEST&
```

whereas

```
nohup cpio -o <list >/dev/rmt/c0t0d0BEST 2>errors&
```

puts the error messages into file **errors**.

**EXIT STATUS**

The following exit values are returned:

**126**      The command specified by *command* was found but could not be invoked

**127**        An error occurred in the nohup utility or the specified *command* could not be found  
Otherwise, the exit status of nohup will be that of the command specified.

**SEE ALSO**

chmod(1), nice(1), sh(1), signal(5).

**STANDARDS CONFORMANCE**

**nohup**: SVID2, SVID3, XPG2, XPG3, XPG4, POSIX.2

| **n** |