



## ***Red Hat Enterprise Linux Release 9.2 Manual Pages on 'getconf.1p' command***

***\$ man getconf.1p***

GETCONF(1P)            POSIX Programmer's Manual            GETCONF(1P)

### PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

### NAME

getconf ? get configuration values

### SYNOPSIS

```
getconf [-v specification] system_var
getconf [-v specification] path_var pathname
```

### DESCRIPTION

In the first synopsis form, the `getconf` utility shall write to the standard output the value of the variable specified by the `system_var` operand.

In the second synopsis form, the `getconf` utility shall write to the standard output the value of the variable specified by the `path_var` operand for the path specified by the `pathname` operand.

The value of each configuration variable shall be determined as if it were obtained by calling the function from which it is defined to be available by this volume of POSIX.1?2017 or by the System Interfaces volume of POSIX.1?2017 (see the OPERANDS section). The value shall reflect conditions in the current operating environment.

## OPTIONS

The `getconf` utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following option shall be supported:

`-v` specification

Indicate a specific specification and version for which configuration variables shall be determined. If this option is not specified, the values returned correspond to an implementation default conforming compilation environment.

If the command:

```
getconf _POSIX_V7_ILP32_OFF32
```

does not write `"-1\n"` or `"undefined\n"` to standard output,

then commands of the form:

```
getconf -v POSIX_V7_ILP32_OFF32 ...
```

determine values for configuration variables corresponding to the `POSIX_V7_ILP32_OFF32` compilation environment specified in c99, the EXTENDED DESCRIPTION.

If the command:

```
getconf _POSIX_V7_ILP32_OFFBIG
```

does not write `"-1\n"` or `"undefined\n"` to standard output,

then commands of the form:

```
getconf -v POSIX_V7_ILP32_OFFBIG ...
```

determine values for configuration variables corresponding to the `POSIX_V7_ILP32_OFFBIG` compilation environment specified in c99, the EXTENDED DESCRIPTION.

If the command:

```
getconf _POSIX_V7_LP64_OFF64
```

does not write `"-1\n"` or `"undefined\n"` to standard output,

then commands of the form:

```
getconf -v POSIX_V7_LP64_OFF64 ...
```

determine values for configuration variables corresponding to the `POSIX_V7_LP64_OFF64` compilation environment specified in c99, the EXTENDED DESCRIPTION.

If the command:

```
getconf _POSIX_V7_LPBIG_OFFBIG
```

does not write "-1\n" or "undefined\n" to standard output,

then commands of the form:

```
getconf -v POSIX_V7_LPBIG_OFFBIG ...
```

determine values for configuration variables corresponding to

the `_POSIX_V7_LPBIG_OFFBIG` compilation environment specified

in c99, the EXTENDED DESCRIPTION.

## OPERANDS

The following operands shall be supported:

`path_var` A name of a configuration variable. All of the variables in the Variable column of the table in the DESCRIPTION of the `fpathconf()` function defined in the System Interfaces volume of POSIX.1?2017, without the enclosing braces, shall be supported. The implementation may add other local variables.

`pathname` A pathname for which the variable specified by `path_var` is to be determined.

`system_var`

A name of a configuration variable. All of the following variables shall be supported:

- \* The names in the Variable column of the table in the DESCRIPTION of the `sysconf()` function in the System Interfaces volume of POSIX.1?2017, except for the entries corresponding to `_SC_CLK_TCK`, `_SC_GETGR_R_SIZE_MAX`, and `_SC_GETPW_R_SIZE_MAX`, without the enclosing braces.

For compatibility with earlier versions, the following

variable names shall also be supported: `POSIX2_C_BIND`

`POSIX2_C_DEV` `POSIX2_CHAR_TERM` `POSIX2_FORT_DEV`

`POSIX2_FORT_RUN` `POSIX2_LOCALEDEF` `POSIX2_SW_DEV` `POSIX2_UPE`

`POSIX2_VERSION`

and shall be equivalent to the same name prefixed with an

<underscore>. This requirement may be removed in a fu?

ture version.

- \* The names of the symbolic constants used as the name argument of the `confstr()` function in the System Interfaces volume of POSIX.1-2017, without the `_CS_` prefix.
- \* The names of the symbolic constants listed under the headings "Maximum Values" and "Minimum Values" in the description of the `<limits.h>` header in the Base Definitions volume of POSIX.1-2017, without the enclosing braces.

For compatibility with earlier versions, the following variable names shall also be supported:

`POSIX2_BC_BASE_MAX` `POSIX2_BC_DIM_MAX` `POSIX2_BC_SCALE_MAX`  
`POSIX2_BC_STRING_MAX` `POSIX2_COLL_WEIGHTS_MAX`  
`POSIX2_EXPR_NEST_MAX` `POSIX2_LINE_MAX` `POSIX2_RE_DUP_MAX`

and shall be equivalent to the same name prefixed with an `<underscore>`. This requirement may be removed in a future version.

The implementation may add other local values.

## STDIN

Not used.

## INPUT FILES

None.

## ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of `getconf`:

**LANG** Provide a default value for the internationalization variables that are unset or null. (See the Base Definitions volume of POSIX.1-2017, Section 8.2, Internationalization Variables for the precedence of internationalization variables used to determine the values of locale categories.)

**LC\_ALL** If set to a non-empty string value, override the values of all the other internationalization variables.

**LC\_CTYPE** Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as

opposed to multi-byte characters in arguments).

## LC\_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error.

NLSPATH Determine the location of message catalogs for the processing of LC\_MESSAGES.

## ASYNCHRONOUS EVENTS

Default.

## STDOUT

If the specified variable is defined on the system and its value is described to be available from the `confstr()` function defined in the System Interfaces volume of POSIX.1-2017, its value shall be written in the following format:

```
"%s\n", <value>
```

Otherwise, if the specified variable is defined on the system, its value shall be written in the following format:

```
"%d\n", <value>
```

If the specified variable is valid, but is undefined on the system, `getconf` shall write using the following format:

```
"undefined\n"
```

If the variable name is invalid or an error occurs, nothing shall be written to standard output.

## STDERR

The standard error shall be used only for diagnostic messages.

## OUTPUT FILES

None.

## EXTENDED DESCRIPTION

None.

## EXIT STATUS

The following exit values shall be returned:

- 0 The specified variable is valid and information about its current state was written successfully.

>0 An error occurred.

## CONSEQUENCES OF ERRORS

Default.

The following sections are informative.

## APPLICATION USAGE

None.

## EXAMPLES

The following example illustrates the value of {NGROUPS\_MAX}:

```
getconf NGROUPS_MAX
```

The following example illustrates the value of {NAME\_MAX} for a specific directory:

```
getconf NAME_MAX /usr
```

The following example shows how to deal more carefully with results that might be unspecified:

```
if value=$(getconf PATH_MAX /usr); then
    if [ "$value" = "undefined" ]; then
        echo PATH_MAX in /usr is indeterminate.
    else
        echo PATH_MAX in /usr is $value.
    fi
else
    echo Error in getconf.
fi
```

## RATIONALE

The original need for this utility, and for the `confstr()` function, was to provide a way of finding the configuration-defined default value for the `PATH` environment variable. Since `PATH` can be modified by the user to include directories that could contain utilities replacing the standard utilities, shell scripts need a way to determine the system-supplied `PATH` environment variable value that contains the correct search path for the standard utilities. It was later suggested that access to the other variables described in this volume of POSIX.1?2017 could also be useful to applications.

This functionality of `getconf` would not be adequately subsumed by another command such as:

```
grep var /etc/conf
```

because such a strategy would provide correct values for neither those variables that can vary at runtime, nor those that can vary depending on the path.

Early proposal versions of `getconf` specified exit status 1 when the specified variable was valid, but not defined on the system. The output string "undefined" is now used to specify this case with exit code 0 because so many things depend on an exit code of zero when an invoked utility is successful.

## FUTURE DIRECTIONS

None.

## SEE ALSO

`c99`

The Base Definitions volume of POSIX.1-2017, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guidelines, `<limits.h>`

The System Interfaces volume of POSIX.1-2017, `confstr()`, `fpathconf()`, `sysconf()`, `system()`

## COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html>.

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see [https://www.kernel.org/doc/man-pages/reporting\\_bugs.html](https://www.kernel.org/doc/man-pages/reporting_bugs.html).

