



Linux Ubuntu 26.04 Manual Pages on command 'pkg-config.1'

\$ man pkg-config.1

PKGCONF(1)

General Commands Manual

PKGCONF(1)

NAME

pkgconf ? a system for configuring build dependency information

SYNOPSIS

pkgconf [options] module ...

DESCRIPTION

The pkgconf program retrieves configuration information related to the module arguments from pc(5) files installed on the system and prints parts of the retrieved information depending on the specified options. The most common use is printing the compiler and linker flags needed to build software that uses the libraries given by the module arguments.

The pc(5) files are searched for along a path constructed from the --with-path option, the PKG_CONFIG_PATH and PKG_CONFIG_LIBDIR environment variables, and some compiled-in default directories. The module arguments correspond to the file names, but without the .pc file? name extension.

Several of the options cause immediate exit. If multiple of these options are given, only the option with the highest priority takes effect and those with lower priority are silently ignored. These options are, ordered by descending priority:

1. No-module options: --relocate, --dump-personality, --about, --version, --help, --atleast-pkgconfig-version, --list-all, and --list-package-names: These options cause all arguments to be ignored.
2. Argument-only options: --atleast-version, --exact-version, and --max-version: These options only inspect modules explicitly specified on the command line and do not look at dependencies.

3. Limited-output options: `--validate`, `--license`, `--uninstalled`, and `--env`: These options perform dependency resolution, but exit after printing the information requested by the highest-priority option, ignoring other output options that may have been specified.

Several other options require at least one module argument, produce output, do not cause early exit, can be combined with each other, but override and disable all `--cflags` and `--libs` options:

1. Single-module output options: `--path`, `--print-variables`, `--variable`: If any of these options is specified, only the first module argument is used, all other arguments are silently ignored, and no dependency resolution is attempted.
2. Depth-one output options: `--print-provides`, `--modversion`, `--print-requires`, and `--print-requires-private`: If any of these options is specified, only modules explicitly specified on the command line are inspected and no dependency resolution is attempted.
3. General output options: `--simulate`, `--digraph`, `--solution`, `--fragment-tree`: These options do not limit dependency resolution.

The most important output options `--cflags` and `--libs` can be combined with each other, but are overridden and ignored if any of the options listed above are specified.

The complete list of options is as follows:

`--about`

Print the version number, the Copyright notice, and the license of the `pkgconf` program to standard output and exit. Most other options and all command line arguments are ignored.

`--atleast-pkgconfig-version=version`

Exit with error if the requested version number is greater than the version number of the `pkgconf` program, or with success otherwise. Most other options and all command line arguments are ignored.

`--atleast-version=version`

Check the module arguments in the given order. Exit with error as soon as a module does not exist, and exit with success as soon as the version number of a module is greater than or equal to the requested version number. Exit with error if the version number of each module is less than the requested version number.

`--cflags`, `--cflags-only-I`, `--cflags-only-other`

Print all compiler flags required to compile against the module, or only the include path (-I) flags, or only the compiler flags that are not include path flags, respectively.

tively. These options imply `--print-errors`.

`--debug`

Print some non-fatal warning messages to standard error output that would otherwise silently be ignored. This option also implies `--print-errors`. If `pkgconf` was compiled without defining the preprocessor macro `PKGCONF_LITE`, this option also prints many debugging messages to standard error output.

`--define-prefix`

Attempts to determine the prefix variable to use for `CFLAGS` and `LIBS` entry relocations. This is mainly useful for platforms where framework SDKs are relocatable, such as Windows.

`--define-variable=varname=value`

Define `varname` as `value`. Variables are used in query output, and some modules' results may change based on the presence of a variable definition.

`--digraph`

Dump the dependency resolver's solution as a graphviz `?dot?` file. This can be used with `graphviz` to visualize module interdependencies. This option is only available if the preprocessor macro `PKGCONF_LITE` was not defined during compilation.

`--dont-define-prefix`

Disables the `?define-prefix?` feature.

`--dont-relocate-paths`

Disables the path relocation feature.

`--dump-personality`

Print some default settings to standard output, in particular the default module search path that is used when `PKG_CONFIG_LIBDIR` is not defined, the default list of include paths that are filtered out when `PKG_CONFIG_SYSTEM_INCLUDE_PATH` is not defined, and the default list of library paths that are filtered out when `PKG_CONFIG_SYSTEM_LIBRARY_PATH` is not defined, and exit. Most other options and all command line arguments are ignored. This option is only available if the preprocessor macro `PKGCONF_LITE` was not defined during compilation.

`--env=varname`

Print the requested values as variable declarations in a similar format as the `env(1)` command.

`--env-only`

Initialize the module search path from --with-path and PKG_CONFIG_PATH only, ignoring PKG_CONFIG_LIBDIR and the compiled-in default directories.

--errors-to-stdout

Print all error, warning, and debugging messages to standard output instead of to standard error output.

--exact-version=version

Check the module arguments in the given order. Exit with error as soon as a module does not exist, and exit with success as soon as the version number of a module is exactly the requested version number. Exit with error if the version number of each module differs from the requested version number.

--exists

Exit with a non-zero exit status if the dependency resolver is unable to find all of the requested modules. This option is active by default and cannot be disabled. However, various other options cause pkgconf to exit and report success or failure before all arguments have been inspected.

--fragment-filter=types

Filter the fragment lists for the specified types.

--help Print a usage summary on standard output and exit. Most other options and all command line arguments are ignored.

--ignore-conflicts

Ignore ?Conflicts? rules in modules.

--keep-system-cflags, --keep-system-libs

Keep CFLAGS or linker flag fragments that would be filtered due to being included by default in the compiler.

--libs, --libs-only-L, --libs-only-l, --libs-only-other

Print all linker flags required to link against the module, or only the library path (-L) flags, or only the library (-l) flags, or only the linker flags that are neither library path nor library flags, respectively. These options imply --print-errors.

--list-all

Walk the module search path in the order of descending priority. For each file found, print one line to standard output, containing the basename of the file without the extension, the Name property, a dash (?-?), and the Description property.

erty. This option implies `--print-errors`. All command line arguments are ignored.

`--list-package-names`

Perform the same search as `--list-all`, but only print the basename of each pc(5) file without the extension, not the module name and the description. This option implies `--print-errors`. All command line arguments are ignored.

`--log-file=file`

Set the name of the output file where information about selected modules is logged, both about those selected by arguments and as dependencies. For each selected module, one line is printed, containing the basename of the pc(5) file without the extension, optionally an operator and version number describing the desired range of versions, and either the actual version number in square brackets or the string "NOT-FOUND". If this option is not provided, the name of the output file is instead taken from the `PKG_CONFIG_LOG` environment variable, and if that is not provided either, this kind of logging is disabled.

`--max-version=version`

Check the module arguments in the given order. Exit with error as soon as a module does not exist, and exit with success as soon as the version number of a module is less than or equal to the requested version number. Exit with error if the version number of each module is greater than the requested version number.

`--maximum-traverse-depth=depth`

Impose a limit on the allowed depth in the dependency graph. For example, a depth of 2 restricts the resolver from acting on child dependencies of modules added to the resolver's solution. This option is overridden by the `PKG_CONFIG_MAXIMUM_TRAVERSE_DEPTH` environment variable and by the options `--modversion`, `--path`, `--print-provides`, `--print-requires`, `--print-requires-private`, `--print-variables`, and `--variable`.

`--modversion`

For each specified module, print the version number to standard output. If the `--verbose` option is also specified, the name of the respective module and a colon is printed before each version number. This option implies `--print-errors` and `--maximum-traverse-depth=1` and overrides and disables all `--cflags` and `--libs` flags.

`--msvc-syntax`

Use MSVC syntax for `--cflags`, `--env`, and `--libs` output. This option is only avail?

able if the preprocessor macro PKGCONF_LITE was not defined during compilation.

--no-cache

Skip caching packages when they are loaded into the internal resolver. This may result in an alternate dependency graph being computed.

--no-provides

Ignore ?Provides? rules in modules when resolving dependencies.

--no-uninstalled

Forbids the dependency resolver from considering 'uninstalled' modules as part of a solution.

--path For the first module given on the command line, let the dependency resolver find the file describing that module, print the absolute pathname of that file to standard output, and exit immediately, ignoring most other options and all other arguments.

--prefix-variable=variable

Sets the ?prefix? variable used by the ?define-prefix? feature.

--print-errors

Print some messages about fatal errors to standard error output that would otherwise be omitted. This option is implied by many other options, but not by all. It can be overridden with --silence-errors.

--print-provides

For each specified module, print one line to standard output containing the Name property, an equal sign (=?), and the Version property. If the module contains one or more Provides properties, print additional lines in dependency list format, one name per line, each name optionally followed by an operator and a version. This option implies --maximum-traverse-depth=1 and overrides and disables all --cflags and --libs flags.

--print-requires, --print-requires-private

For each specified module, print the Requires or Requires.private properties, respectively, in dependency list format to standard output. Both of these options imply --maximum-traverse-depth=1 and override and disable all --cflags and --libs flags.

--print-variables

For the first module given on the command line, print the names of all seen variables.

ables to standard output, one per line. Any subsequent arguments are silently ignored. This option implies `--print-errors` and `--maximum-traverse-depth=1` and overrides and disables all `--cflags` and `--libs` flags.

`--pure` Treats the computed dependency graph as if it were pure. This is mainly intended for use with the `--static` flag and has no effect if `--shared` is also specified.

`--relocate=path`

Relocates a path using the `pkgconf_path_relocate` API. This is mainly used by the testsuite to provide a guaranteed interface to the system's path relocation backend.

`--shared`

Compute a simple dependency graph that is only suitable for shared linking. This option overrides `--static`.

`--short-errors`

When printing error messages about modules that are not found or conflict with each other, avoid printing additional, verbose instructions explaining potential methods for solving the problem.

`--silence-errors`

Do not print any error, warning, or debugging messages at all. Overrides all of `--debug`, `--errors-to-stdout`, and `--print-errors`. This option is overridden and disabled if the `PKG_CONFIG_DEBUG_SPEW` environment variable is set.

`--simulate`

Simulates resolving a dependency graph based on the requested modules on the command line. Dumps a series of trees denoting `pkgconf`'s resolver state. This option is only available if the preprocessor macro `PKGCONF_LITE` was not defined during compilation.

`--solution`

Print the names of the modules requested with module arguments and their dependencies to standard output. This option is only available if the preprocessor macro `PKGCONF_LITE` was not defined during compilation.

`--static`

Compute a deeper dependency graph and use compiler/linker flags intended for static linking. This option is overridden by `--shared`.

`--uninstalled`

Exit with a non-zero result if the dependency resolver uses an `uninstalled?` module

as part of its solution.

--validate package ...

Validate specific `?.pc?` files for correctness. This option implies `--print-errors` and `--errors-to-stdout`.

--variable=varname

For the first module given on the command line, print the value of the variable with the name `varname` to standard output. Any subsequent arguments are silently ignored. This option implies `--maximum-traverse-depth=1` and overrides and disables all `--cflags` and `--libs` flags.

--verbose

This option only has an effect if `--modversion` is also specified. It prints the name of the respective module and a colon before each version number.

--version

Print the version number of the `pkgconf` program to standard output and exit. Most other options and all command line arguments are ignored.

--with-path=path

Prepend the directory `path` to the module search path, giving it priority over all other directories including those from `PKG_CONFIG_PATH` and `PKG_CONFIG_LIBDIR`.

ENVIRONMENT

CPATH First supplementary colon-separated list of include paths filtered out in the same way as `PKG_CONFIG_SYSTEM_INCLUDE_PATH`.

CPLUS_INCLUDE_PATH

Third supplementary colon-separated list of include paths filtered out in the same way as `PKG_CONFIG_SYSTEM_INCLUDE_PATH`.

C_INCLUDE_PATH

Second supplementary colon-separated list of include paths filtered out in the same way as `PKG_CONFIG_SYSTEM_INCLUDE_PATH`.

DESTDIR

If set to the same value as `PKG_CONFIG_SYSROOT_DIR`, behave in the same way as if `PKG_CONFIG_FDO_SYSROOT_RULES` is set. If `PKG_CONFIG_SYSROOT_DIR` is not set or set to a different value, `DESTDIR` is ignored.

LIBRARY_PATH

Supplementary colon-separated list of library paths filtered out in the same way as

PKG_CONFIG_SYSTEM_LIBRARY_PATH.

OBJC_INCLUDE_PATH

Fourth supplementary colon-separated list of include paths filtered out in the same way as PKG_CONFIG_SYSTEM_INCLUDE_PATH.

PKG_CONFIG_ALLOW_SYSTEM_CFLAGS

If set, this variable has the same effect as the `--keep-system-cflags` option.

PKG_CONFIG_ALLOW_SYSTEM_LIBS

If set, this variable has the same effect as the `--keep-system-libs` option.

PKG_CONFIG_DEBUG_SPEW

If set, override and disable the `--silence-errors` option.

PKG_CONFIG_DISABLE_UNINSTALLED

If set, enables the same behaviour as the `--no-uninstalled` flag.

PKG_CONFIG_DONT_DEFINE_PREFIX

If set, this variable has the same effect as the `--dont-define-prefix` option.

PKG_CONFIG_DONT_RELOCATE_PATHS

If set, disables the path relocation feature.

PKG_CONFIG_FDO_SYSROOT_RULES

If set, follow the sysroot prefixing rules that freedesktop.org pkg-config uses.

PKG_CONFIG_IGNORE_CONFLICTS

If set, ignore Conflicts rules in modules. Has the same effect as the `--ignore-conflicts` option.

PKG_CONFIG_LIBDIR

A colon-separated list of low-priority directories where `pc(5)` files are looked up.

The module search path is constructed by appending this list to `PKG_CONFIG_PATH`, which enjoys higher priority. If `PKG_CONFIG_LIBDIR` is not defined, the default list compiled into the `pkgconf` program from the `PKG_DEFAULT_PATH` preprocessor macro is appended instead. If `PKG_CONFIG_LIBDIR` is defined but empty, nothing is appended.

PKG_CONFIG_LOG

If set, log information about selected modules to the file with the name stored in this variable. For more details, see the `--log-file` command line option, which overrides this variable.

PKG_CONFIG_MAXIMUM_TRAVERSE_DEPTH

Impose a limit on the allowed depth in the dependency graph. This variable over?

rides the `--maximum-traverse-depth` option, but is overridden by the other options mentioned there.

PKG_CONFIG_MSVC_SYNTAX

If set, use MSVC syntax for `--cflags`, `--env`, and `--libs` output. This variable has the same effect as the `--msvc-syntax` option. If the preprocessor macro `PKGCONF_LITE` was defined during compilation, this variable is ignored.

PKG_CONFIG_PATH

A colon-separated list of high-priority directories where `pc(5)` files are looked up. The module search path is constructed by prepending the directory specified with `--with-path`, if any, and unless `--env-only` is specified, by appending either `PKG_CONFIG_LIBDIR` or the compiled-in default directories with lower priority.

PKG_CONFIG_PRELOADED_FILES

Colon-separated list of `pc(5)` files which are loaded before any other `pkg-config` files. These packages are given highest priority over any other `pc(5)` files that would otherwise provide a given package.

PKG_CONFIG_PURE_DEPGRAPH

If set, enables the same behaviour as the `--pure` flag.

PKG_CONFIG_RELOCATE_PATHS

If set, this variable has the same effect as the `--define-prefix` option.

PKG_CONFIG_SYSROOT_DIR

If set, this variable defines a `?sysroot?` directory, which will be prepended to every path variable beginning with the prefix variable in a given `pc(5)` file. Useful for cross compilation. The value of this environment variable is also copied into the global variable `pc_sysrootdir`.

PKG_CONFIG_SYSTEM_INCLUDE_PATH

Colon-separated list of include paths that are filtered out and not printed by the `--cflags` and `--cflags-only-I` options because they are considered system include paths. If not defined, the default list compiled into the `pkgconf` program from the `SYSTEM_INCLUDEDIR` preprocessor macro is used instead. This variable is a `pkgconf`-specific extension. Any directories listed in the environment variables `CPATH`, `C_INCLUDE_PATH`, `CPLUS_INCLUDE_PATH`, and `OBJC_INCLUDE_PATH` are also filtered out.

PKG_CONFIG_SYSTEM_LIBRARY_PATH

Colon-separated list of library paths that are filtered out and not printed by the

--libs and --libs-only-L options because they are considered system library paths.

If not defined, the default list compiled into the pkgconf program from the SYSTEM_LIBDIR preprocessor macro is used instead. This variable is a pkgconf-specific extension.

PKG_CONFIG_TOP_BUILD_DIR

The value of the pc_top_builddir global variable. If this environment variable is not defined, the string "\$(top_builddir)" is used as the value of pc_top_builddir.

EXIT STATUS

The pkgconf utility exits 0 on success, and >0 if an error occurs.

EXAMPLES

Displaying the CFLAGS of a package:

```
$ pkgconf --cflags foo
```

```
-fPIC -I/usr/include/foo
```

SEE ALSO

pc(5), pkg.m4(7)

Debian

November 15, 2016

PKGCONF(1)