

## NAME

dpkg-deb - Debian package archive (.deb) manipulation tool

## SYNOPSIS

**dpkg-deb** [*option...*] *command*

## DESCRIPTION

**dpkg-deb** packs, unpacks and provides information about Debian archives.

Use **dpkg** to install and remove packages from your system.

You can also invoke **dpkg-deb** by calling **dpkg** with whatever options you want to pass to **dpkg-deb**. **dpkg** will spot that you wanted **dpkg-deb** and run it for you.

## COMMANDS

**-b, --build** *directory* [*archive|directory*]

Creates a debian archive from the filesystem tree stored in *directory*. *directory* must have a **DEBIAN** subdirectory, which contains the control information files such as the control file itself. This directory will *not* appear in the binary package's filesystem archive, but instead the files in it will be put in the binary package's control information area.

Unless you specify **--nocheck**, **dpkg-deb** will read **DEBIAN/control** and parse it. It will check it for syntax errors and other problems, and display the name of the binary package being built. **dpkg-deb** will also check the permissions of the maintainer scripts and other files found in the **DEBIAN** control information directory.

If no *archive* is specified then **dpkg-deb** will write the package into the file *directory.deb*.

If the archive to be created already exists it will be overwritten.

If the second argument is a directory then **dpkg-deb** will write to the file *package\_version\_arch.deb*, or *package\_version.deb* if no **Architecture** field is present in the package control file. When a target directory is specified, rather than a file, the **--nocheck** option may not be used (since **dpkg-deb** needs to read and parse the package control file to determine which filename to use).

**-I, --info** *archive* [*control-file-name...*]

Provides information about a binary package archive.

If no *control-file-names* are specified then it will print a summary of the contents of the package as well as its control file.

If any *control-file-names* are specified then **dpkg-deb** will print them in the order they were specified; if any of the components weren't present it will print an error message to stderr about each one and exit with status 2.

**-W, --show** *archive*

Provides information about a binary package archive in the format specified by the **--showformat** argument. The default format displays the package's name and version on one line, separated by a tabulator.

**-f, --field** *archive* [*control-field-name...*]

Extracts control file information from a binary package archive.

If no **control-file-fields** are specified then it will print the whole control file.

If any are specified then **dpkg-deb** will print their contents, in the order in which they appear in the control file. If more than one **control-file-field** is specified then **dpkg-deb** will precede each with its field name (and a colon and space).

No errors are reported for fields requested but not found.

**-c, --contents** *archive*

Lists the contents of the filesystem tree archive portion of the package archive. It is currently produced in the format generated by **tar**'s verbose listing.

**-x, --extract** *archive directory*

Extracts the filesystem tree from a package archive into the specified directory.

Note that extracting a package to the root directory will *not* result in a correct installation! Use **dpkg** to install packages.

*directory* (but not its parents) will be created if necessary, and its permissions modified to match the contents of the package.

**-X, --vextract** *archive directory*

Is like **--extract** (**-x**) with **--verbose** (**-v**) which prints a listing of the files extracted as it goes.

**-R, --raw-extract** *archive directory*

Extracts the filesystem tree from a package archive into a specified directory, and the control information files into a DEBIAN subdirectory of the specified directory.

The target directory (but not its parents) will be created if necessary.

**--ctrl-tarfile** *archive*

Extracts the control data from a binary package and sends it to standard output in **tar** format (since dpkg 1.17.14). Together with **tar(1)** this can be used to extract a particular control file from a package archive. The input archive will always be processed sequentially.

**--fsys-tarfile** *archive*

Extracts the filesystem tree data from a binary package and sends it to standard output in **tar** format. Together with **tar(1)** this can be used to extract a particular file from a package archive. The input archive will always be processed sequentially.

**-e, --control** *archive [directory]*

Extracts the control information files from a package archive into the specified directory.

If no directory is specified then a subdirectory **DEBIAN** in the current directory is used.

The target directory (but not its parents) will be created if necessary.

**-?, --help**

Show the usage message and exit.

**--version**

Show the version and exit.

**OPTIONS****--showformat=***format*

This option is used to specify the format of the output **--show** will produce. The format is a string that will be output for each package listed.

The string may reference any status field using the  $\${field-name}$  form, a list of the valid fields can be easily produced using **-I** on the same package. A complete explanation of the formatting options (including escape sequences and field tabbing) can be found in the explanation of the **--showformat** option in **dpkg-query(1)**.

The default for this field is  $\${Package}t${Version}n$ .

**-z***compress-level*

Specify which compression level to use on the compressor backend, when building a package (default is 9 for gzip and bzip2, 6 for xz and lzma). The accepted values are 0-9 with: 0 being mapped to compressor none for gzip and 0 mapped to 1 for bzip2. Before dpkg 1.16.2 level 0 was equivalent to compressor none for all compressors.

**-S** *compress-strategy*

Specify which compression strategy to use on the compressor backend, when building a package (since dpkg 1.16.2). Allowed values are *none* (since dpkg 1.16.4), *filtered*, *huffman*, *rle* and *fixed* for gzip (since dpkg 1.17.0) and *extreme* for xz.

**-Z** *compress-type*

Specify which compression type to use when building a package. Allowed values are *gzip*, *xz*, *bzip2* (deprecated), *lzma* (deprecated), and *none* (default is *xz*).

**--uniform-compression**

Specify that the same compression parameters should be used for all archive members (i.e. **control.tar** and **data.tar**). Otherwise only the **data.tar** member will use those parameters. The only supported compression types allowed to be uniformly used are *none*, *gzip* and *xz*.

**--deb-format=***format*

Set the archive format version used when building (since dpkg 1.17.0). Allowed values are *2.0* for the new format, and *0.939000* for the old one (default is *2.0*).

The old archive format is less easily parsed by non-Debian tools and is now obsolete; its only use is when building packages to be parsed by versions of dpkg older than 0.93.76 (September 1995), which was released as i386 a.out only.

**--new** This is a legacy alias for **--deb-format=2.0**.

**--old** This is a legacy alias for **--deb-format=0.939000**.

**--nocheck**

Inhibits **dpkg-deb --build**'s usual checks on the proposed contents of an archive. You can build any archive you want, no matter how broken, this way.

**-v, --verbose**

Enables verbose output. This currently only affects **--extract** making it behave like **--vextract**.

**-D, --debug**

Enables debugging output. This is not very interesting.

**ENVIRONMENT****TMPDIR**

If set, **dpkg-deb** will use it as the directory in which to create temporary files and directories.

**NOTES**

Do not attempt to use just **dpkg-deb** to install software! You must use **dpkg** proper to ensure that all the files are correctly placed and the package's scripts run and its status and contents recorded.

**BUGS**

**dpkg-deb -I package1.deb package2.deb** does the wrong thing.

There is no authentication on **.deb** files; in fact, there isn't even a straightforward checksum. (Higher level tools like APT support authenticating **.deb** packages retrieved from a given repository, and most packages nowadays provide an md5sum control file generated by debian/rules.) Though this is not directly supported by the lower level tools.)

**SEE ALSO**

[deb\(5\)](#), [deb-control\(5\)](#), [dpkg\(1\)](#), [dselect\(1\)](#).