NAME

dput − Debian package upload tool

SYNOPSIS

dput [−**DPUVdflosu**] [−**c** *CONFIGFILE*] [−**e** *DAYS*] [*HOSTNAME*[:ARGUMENT]*] CHANGESFILE* . . .

dput [−**d**] [−**c** *CONFIGFILE*] −**H**

dput [−**d**] [−**c** *CONFIGFILE*] −**p**

dput −**h**

dput −**v**

DESCRIPTION

Upload, to the Debian package upload queue, the files constituting the package specified in each *CHANGESFILE*.

A Debian package release comprises several files, specified in the *CHANGESFILE* for that release. The Debian Package Upload Tool (**dput**) performs several verification checks on the package, then (if the verifications succeed) uploads those files to the remote Debian package repository. See **VERIFICATION CHECKS**, below, for a description of the checks **dput** can perform on each package.

HOSTNAME is a name matching a host section in the **dput.cf**(5) configuration. If not specified, *HOSTNAME* defaults to the value of the **default_host_main** configuration parameter. You also can pass an argument to the host by appending the hostname with a colon followed by the argument. For example, you can upload easily to different launchpad PPAs by passing the PPAs' id to the ppa host without having to modify dput.cf.

The file transfer method is determined by the **method** configuration parameter for the specified host. See **dput.cf**(5) for the set of supported **method** values.

If the configuration specifies a **pre_upload_command** for the remote host, execute a shell subprocess for that command before attempting to upload each package.

If the configuration specifies a **post_upload_command** for the remote host, execute a shell subprocess for that command after each successful package upload.

VERIFICATION CHECKS

Each package release, as specified by a *CHANGESFILE*, is checked to see whether it is suitable for upload. The following checks are available:

Unique upload

> The package should not already have been uploaded to the remote host *HOSTNAME*.

> Check for the log file created when **dput** successfully uploads a package. If the corresponding log file already exists, abort the upload attempt. This is unique only for the combination of local machine (where the upload log is recorded), the *CHANGESFILE* name, and the remote *HOSTNAME*.

> This check is disabled if the **−−force** option is specified.

Required package files

> The package should comprise the required files for the type of package.

> If this is a source package release, and the required source control file (*….dsc*) is not part of the package, abort the upload attempt.

> If this is a source package release, and the version string implies that this is the first Debian release for this version, and the corresponding source tarball is not part of the package, abort the upload attempt.

> If the package contains the upstream source tarball, but this is not required for this release, emit a warning that the upload might be rejected by the remote host.

> If the package contains the Debian-native source tarball, but this is not required for this release, emit a warning that the upload might be rejected by the remote host.

Successful local install

>   The package should be successfully installed on the local system.
>
>   If this is a binary package release, and each binary package (*.…deb* file) is not successfully in-stalled on the local system at the specified version, abort the upload.
>
>   This check is disabled by default. It is enabled when the configuration parameter **check_version** is true.

Cryptographic signature

>   The GnuPG signature on the *CHANGESFILE* should match that file's contents. If this is a source package release, the GnuPG signature on the source control file (*.…dsc*) should match that file's contents.
>
>   If GnuPG does not report a good signature for these files, abort the upload.
>
>   This check is disabled when either the **allow_unsigned_uploads** parameter is true, or the **−−check−only** option is specified.

Package files content match

>   The computed size and hash values for each constituent file should match the values specified in *CHANGESFILE*.
>
>   For each file size and hash value specified in *CHANGESFILE*, compute the size and hash using the corresponding named file's contents on the filesystem. If any computed size or hash value does not match the corresponding value in *CHANGESFILE*, abort the upload.

Distribution name is permitted for the remote host

>   The configuration for *HOSTNAME* should permit the distribution name specified in the package.
>
>   If the package release's target Debian distribution name is not one specified by the **distributions** parameter for the *HOSTNAME*, abort the upload attempt. See **dput.cf**(5) for the **distributions** parameter.

Lintian success

>   The package should get a successful report from Lintian.
>
>   Invoke the Lintian tool to check the package described by *CHANGESFILE*. If the check is unsuc-cessful (exit status non-zero), abort the upload attempt.
>
>   This check is disabled by default. It is enabled when the configuration parameter **run_lintian** is true, or the **−−lintian** option is specified.

OPTIONS

−c *CONFIGFILE*
−−config *CONFIGFILE*
>   Use the file *CONFIGFILE* for configuration. See **dput.cf**(5) for the specification of this file for-mat.

−d
−−debug
>   Display debugging messages.

−D
−−dinstall
>   Request a dry run of **dinstall** on the remote host after the upload.

−f
−−force
>   Disable the check for a unique upload. See **VERIFICATION CHECKS**, above, for a description of this check.

−h
−−help   Display help information, then exit.

−H
−−host−list
>   Display the lists of hosts known to the configuration, then exit. See **dput.cf**(5) for the specification of hosts.

−l
−−lintian
>   Enable the Lintian verification check, and exit if the check fails. See **VERIFICATION CHECKS**, above, for a description of this check.

−U
−−no−upload−log
>   Do not write a log file when uploading files.
>
>   This omits the record used later for checking whether an upload is unique. See **VERIFICATION CHECKS**, above, for a description of this check.

−o
−−check−only
>   Do not upload the files, and disable the checks for unique upload and cryptographic signature. See **VERIFICATION CHECKS**, above, for a description of these checks.

−p
−−print
>   Display the configuration that **dput** is using, then exit. See **dput.cf**(5) for the configuration format.

−P
−−passive
>   Use passive mode for FTP. If not specified, default for FTP is active mode.

−s
−−simulate
>   Do not actually upload to the remote host.

−u
−−unchecked
>   Disable the check for cryptographic signature. See **VERIFICATION CHECKS**, above, for a description of this check.

−e *DAYS*
−−delayed *DAYS*
>   Upload to a **DELAYED** queue, rather than the usual **Incoming**. This takes a *DAYS* argument from 0 to 15. Note that a delay of 0 is different from no delay at all.

−v
−−version
>   Display version information, then exit.

−V
−−check−version
>   Enable the check for successful local install. See **VERIFICATION CHECKS**, above, for a description of this check.

ENVIRONMENT
> **USER**   If no username is specified for authenticating to the remote host, use the value of this variable.

FILES
> Configuration
>   See **dput.cf**(5) for the files to configure **dput.**

path/*basename*.*HOSTNAME*.**upload**
>   The log file that contains a record of the successful upload to *HOSTNAME*.
>
>   *path* is the directory path of *CHANGESFILE*.
>
>   *basename* is derived from the *CHANGESFILE* filename, by removing the directory path and the **.changes** suffix.

SEE ALSO

dput.cf(5), **dcut**(1), **lintian**(1)

'dput' package documentation ⟨file:///usr/share/doc/dput/⟩, 'dput' program files ⟨file:///usr/share/dput/⟩.