rpm(8) - Linux man page

Name

rpm - RPM Package Manager

Synopsis

Querying and Verifying Packages:

rpm {**-q|--query**} [**select-options**] [**query-options**]

rpm {**-V|--verify**} [**select-options**] [**verify-options**]

rpm --import ***PUBKEY*** *...*

rpm {**-K|--checksig**} [**--nosignature**] [**--nodigest**] ***PACKAGE_FILE*** *...*

Installing, Upgrading, and Removing Packages:

rpm {**-i|--install**} [**install-options**] ***PACKAGE_FILE*** *...*

rpm {**-U|--upgrade**} [**install-options**] ***PACKAGE_FILE*** *...*

rpm {**-F|--freshen**} [**install-options**] ***PACKAGE_FILE*** *...*

rpm {**-e|--erase**} [**--allmatches**] [**--nodeps**] [**--noscripts**] [**--notriggers**] [**--test**] ***PACKAGE_NAME*** *...*

Miscellaneous:

rpm {**--initdb|--rebuilddb**}

rpm {**--addsign|--resign**} ***PACKAGE_FILE****...*

rpm {**--querytags|--showrc**}

rpm {**--setperms|--setugids**} ***PACKAGE_NAME*** *...*

select-options

[***PACKAGE_NAME***] [**-a,--all**] [**-f,--file** ***FILE***]
[**-g,--group** ***GROUP***] {**-p,--package** ***PACKAGE_FILE***]
[**--fileid** ***ID***] [**--hdrid** ***SHA1***] [**--pkgid** ***MD5***] [**--tid** ***TID***]
[**--querybynumber** ***HDRNUM***] [**--triggeredby** ***PACKAGE_NAME***]
[**--whatprovides** ***CAPABILITY***] [**--whatrequires** ***CAPABILITY***]

query-options

[**--changelog**] [**-c,--configfiles**] [**-d,--docfiles**] [**--dump**]
[**--filesbypkg**] [**-i,--info**] [**--last**] [**-l,--list**]
[**--provides**] [**--qf,--queryformat** ***QUERYFMT***]
[**-R,--requires**] [**--scripts**] [**-s,--state**]
[**--triggers,--triggerscripts**]

verify-options

[**--nodeps**] [**--nofiles**] [**--noscripts**]
[**--nodigest**] [**--nosignature**]
[**--nolinkto**] [**--nofiledigest**] [**--nosize**] [**--nouser**]
[**--nogroup**] [**--nomtime**] [**--nomode**] [**--nordev**]
[**--nocaps**]

install-options

[**--aid**] [**--allfiles**] [**--badreloc**] [**--excludepath** ***OLDPATH***]
[**--excludedocs**] [**--force**] [**-h,--hash**]
[**--ignoresize**] [**--ignorearch**] [**--ignoreos**]
[**--includedocs**] [**--justdb**] [**--nodeps**]
[**--nodigest**] [**--nosignature**] [**--nosuggest**]
[**--noorder**] [**--noscripts**] [**--notriggers**]
[**--oldpackage**] [**--percent**] [**--prefix** ***NEWPATH***]
[**--relocate** ***OLDPATH=NEWPATH***]
[**--replacefiles**] [**--replacepkgs**]
[**--test**]

Description

rpm is a powerful **Package Manager**, which can be used to build, install, query, verify, update, and erase individual software packages. A **package** consists of an archive of files and meta-data used to install and erase the archive files. The meta-data includes helper scripts, file attributes, and descriptive information about the package. **Packages** come in two varieties: binary packages, used to encapsulate software to be installed, and source packages, containing the source code and recipe necessary to produce binary packages.

One of the following basic modes must be selected: **Query**, **Verify**, **Signature Check**, **Install/Upgrade/Freshen**, **Uninstall**, **Initialize Database**, **Rebuild Database**, **Resign**, **Add Signature**, **Set Owners/Groups**, **Show Querytags**, and **Show Configuration**.

General Options

These options can be used in all the different modes.

-?, --help
Print a longer usage message then normal.

--version
Print a single line containing the version number of **rpm** being used.

--quiet
Print as little as possible - normally only error messages will be displayed.

-v
Print verbose information - normally routine progress messages will be displayed.

-vv
Print lots of ugly debugging information.

--rcfile *FILELIST*
Each of the files in the colon separated *FILELIST* is read sequentially by **rpm** for configuration information. Only the first file in the list must exist, and tildes will be expanded to the value of **$HOME**. The default *FILELIST* is */usr/lib/rpm/rpmrc*:*/usr/lib/rpm/redhat/rpmrc*:*/etc/rpmrc*:*~/.rpmrc*.

--pipe *CMD*
Pipes the output of **rpm** to the command *CMD*.

--dbpath *DIRECTORY*
Use the database in *DIRECTORY* rather than the default path */var/lib/rpm*

--root *DIRECTORY*
Use the file system tree rooted at *DIRECTORY* for all operations. Note that this means the database within *DIRECTORY* will be used for dependency checks and any **scriptlet**(s) (e.g. **%post** if installing, or **%prep** if building, a package) will be run after a ***chroot***(*2*) to *DIRECTORY*.

-D, --define='*MACRO EXPR*'
Defines *MACRO* with value *EXPR*.

-E, --eval='*EXPR*'
Prints macro expansion of *EXPR*.

Install and Upgrade Options

In these options, *PACKAGE_FILE* can be either **rpm** binary file or ASCII package manifest (see **PACKAGE SELECTION OPTIONS**), and may be specified as an **ftp** or ***http*** URL, in which case the package will be downloaded before being installed. See **FTP/HTTP OPTIONS** for information on **rpm**'s internal **ftp** and ***http*** client support.

The general form of an rpm install command is

rpm {**-i|--install**} [**install-options**] ***PACKAGE_FILE*** *...*

This installs a new package.

The general form of an rpm upgrade command is

rpm {**-U|--upgrade**} [**install-options**] ***PACKAGE_FILE*** *...*

This upgrades or installs the package currently installed to a newer version. This is the same as install, except all other **version**(s) of the package are removed after the new package is installed.

rpm {**-F|--freshen**} [**install-options**] ***PACKAGE_FILE*** *...*

This will upgrade packages, but only ones for which an earlier version is installed.

--aid
: Add suggested packages to the transaction set when needed.

--allfiles
: Installs or upgrades all the missingok files in the package, regardless if they exist.

--badreloc
: Used with **--relocate**, permit relocations on all file paths, not just those *OLDPATH*'s included in the binary package relocation **hint**(s).

--excludepath *OLDPATH*
: Don't install files whose name begins with *OLDPATH*.

--excludedocs
: Don't install any files which are marked as documentation (which includes man pages and texinfo documents).

--force
: Same as using **--replacepkgs**, **--replacefiles**, and **--oldpackage**.

-h, --hash
: Print 50 hash marks as the package archive is unpacked. Use with **-v|--verbose** for a nicer display.

--ignoresize
: Don't check mount file systems for sufficient disk space before installing this package.

--ignorearch
: Allow installation or upgrading even if the architectures of the binary package and host don't match.

--ignoreos
: Allow installation or upgrading even if the operating systems of the binary package and host don't match.

--includedocs
: Install documentation files. This is the default behavior.

--justdb
: Update only the database, not the filesystem.

--nodigest
: Don't verify package or header digests when reading.

--nomanifest
: Don't process non-package files as manifests.

--nosignature

Don't verify package or header signatures when reading.

--nodeps

Don't do a dependency check before installing or upgrading a package.

--nosuggest

Don't suggest **package**(s) that provide a missing dependency.

--noorder

Don't reorder the packages for an install. The list of packages would normally be reordered to satisfy dependencies.

--noscripts
--nopre
--nopost
--nopreun
--nopostun

Don't execute the scriptlet of the same name. The **--noscripts** option is equivalent to

--nopre --nopost --nopreun

-

-

nopostun

and turns off the execution of the corresponding **%pre**, **%post**, **%preun**, and **%postun scriptlet**(s).

--notriggers
--notriggerin
--notriggerun
--notriggerpostun

Don't execute any trigger scriptlet of the named type. The **--notriggers** option

is equivalent to

--notriggerin --notriggerun --notriggerpostun

and turns off execution of the corresponding **%triggerin**, **%triggerun**, and **%triggerpostun scriptlet**(s).

--oldpackage

Allow an upgrade to replace a newer package with an older one.

--percent

Print percentages as files are unpacked from the package archive. This is intended to make **rpm** easy to run from other tools.

--prefix *NEWPATH*

For relocatable binary packages, translate all file paths that start with the installation prefix in the package relocation **hint**(s) to *NEWPATH*.

--relocate *OLDPATH*=*NEWPATH*

For relocatable binary packages, translate all file paths that start with *OLDPATH* in the package relocation **hint**(s) to *NEWPATH*. This option can be used repeatedly if several *OLDPATH*'s in the package are to be relocated.

--replacefiles

Install the packages even if they replace files from other, already installed, packages.

--replacepkgs

Install the packages even if some of them are already installed on this system.

--test

Do not install the package, simply check for and report potential conflicts.

Erase Options

The general form of an rpm erase command is

rpm {**-e|--erase**} [**--allmatches**] [**--nodeps**] [**--noscripts**] [**--notriggers**] [**--test**] ***PACKAGE_NAME***...

The following options may also be used:

--allmatches

Remove all versions of the package which match *PACKAGE_NAME*. Normally an error is issued if *PACKAGE_NAME* matches multiple packages.

--nodeps

Don't check dependencies before uninstalling the packages.

--noscripts

--nopreun

--nopostun

Don't execute the scriptlet of the same name. The **--noscripts** option during package erase is equivalent

to

--nopreun --nopostun

and turns off the execution of the corresponding **%preun**, and **%postun** **scriptlet**(s).

--notriggers

--notriggerun

--notriggerpostun

Don't execute any trigger scriptlet of the named type. The **--notriggers** option

is equivalent to

--notriggerun --notriggerpostun

and turns off execution of the corresponding **%triggerun**, and **%triggerpostun scriptlet**(s).

--test
Don't really uninstall anything, just go through the motions. Useful in conjunction with the **-vv** option for debugging.

Query Options

The general form of an rpm query command is

rpm {**-q|--query**} [**select-options**] [**query-options**]

You may specify the format that package information should be printed in. To do this, you use the

--qf|--queryformat ***QUERYFMT***

option, followed by the *QUERYFMT* format string. Query formats are modified versions of the standard ***printf(3)*** formatting. The format is made up of static strings (which may include standard C character escapes for newlines, tabs, and other special characters) and ***printf(3)*** type formatters. As **rpm** already knows the type to print, the type specifier must be omitted however, and replaced by the name of the header tag to be printed, enclosed by **{}** characters. Tag names are case insensitive, and the leading **RPMTAG_** portion of the tag name may be omitted as well.

Alternate output formats may be requested by following the tag with **:*typetag***. Currently, the following types are supported:

:armor
Wrap a public key in ASCII armor.

:arraysize
Display number of elements in array tags.

:base64
Encode binary data using base64.

:date
Use ***strftime***(*3*) "%c" format.

:day
Use ***strftime***(*3*) "%a %b %d %Y" format.

:depflags
Format dependency comparison operator.

:deptype
Format dependency type.

:fflags

Format file flags.

:fstate
Format file state.

:hex
Format in hexadecimal.

:octal
Format in octal.

:perms
Format file permissions.

:pgpsig
Display signature fingerprint and time.

:shescape
Escape single quotes for use in a script.

:triggertype
Display trigger suffix.

:vflags
File verification flags.

:xml
Wrap data in simple xml markup.

For example, to print only the names of the packages queried, you could use **%{NAME}** as the format string. To print the packages name and distribution information in two columns, you could use **%-30{NAME}%{DISTRIBUTION}**. **rpm** will print a list of all of the tags it knows about when it is invoked with the **--querytags** argument.

There are two subsets of options for querying: package selection, and information selection.

Package Selection Options:

PACKAGE_NAME
Query installed package named *PACKAGE_NAME*.

-a, --all
Query all installed packages.

-f, --file *FILE*
Query package owning *FILE*.

--fileid *ID*
Query package that contains a given file identifier. The *ID* is the digest of the file contents. For different packages different hash algorithms may have been used (*MD5*, *SHA1*, *SHA256*, *SHA384*, *SHA512*, ...)

-g, --group *GROUP*
Query packages with the group of *GROUP*.

--hdrid *SHA1*

Query package that contains a given header identifier, i.e. the *SHA1* digest of the immutable header region.

-p, --package *PACKAGE_FILE*

Query an (uninstalled) package *PACKAGE_FILE*. The *PACKAGE_FILE* may be specified as an **ftp** or ***http*** style URL, in which case the package header will be downloaded and queried. See **FTP/HTTP OPTIONS** for information on **rpm**'s internal **ftp** and ***http*** client support. The *PACKAGE_FILE* **argument**(s), if not a binary package, will be interpreted as an ASCII package manifest unless **--nomanifest** option is used. In manifests, comments are permitted, starting with a '#', and each line of a package manifest file may include white space separated glob expressions, including URL's, that will be expanded to paths that are substituted in place of the package manifest as additional *PACKAGE_FILE* arguments to the query.

--pkgid *MD5*

Query package that contains a given package identifier, i.e. the *MD5* digest of the combined header and payload contents.

--querybynumber *HDRNUM*

Query the *HDRNUM*th database entry directly; this is useful only for debugging.

--specfile *SPECFILE*

Parse and query *SPECFILE* as if it were a package. Although not all the information (e.g. file lists) is available, this type of query permits rpm to be used to extract information from spec files without having to write a specfile parser.

--tid *TID*

Query **package**(s) that have a given *TID* transaction identifier. A unix time stamp is currently used as a transaction identifier. All **package**(s) installed or erased within a single transaction have a common identifier.

--triggeredby *PACKAGE_NAME*

Query packages that are triggered by **package**(s) *PACKAGE_NAME*.

--whatprovides *CAPABILITY*

Query all packages that provide the *CAPABILITY* capability.

--whatrequires *CAPABILITY*

Query all packages that require *CAPABILITY* for proper functioning.

Package Query Options:

--changelog

Display change information for the package.

-c, --configfiles

List only configuration files (implies **-l**).

-d, --docfiles

List only documentation files (implies **-l**).

--dump

Dump file information as follows (implies **-l**):

```
        path size mtime digest mode owner group isconfig isdoc rdev symlink
```

--filesbypkg
List all the files in each selected package.

-i, --info
Display package information, including name, version, and description. This uses the **--queryformat** if one was specified.

--last
Orders the package listing by install time such that the latest packages are at the top.

-l, --list
List files in package.

--provides
List capabilities this package provides.

-R, --requires
List capabilities on which this package depends.

--scripts
List the package specific **scriptlet**(s) that are used as part of the installation and uninstallation processes.

-s, --state
Display the *states* of files in the package (implies **-l**). The state of each file is one of *normal*, *not installed*, or *replaced*.

--triggers, --triggerscripts
Display the trigger scripts, if any, which are contained in the package.

Verify Options

The general form of an rpm verify command is

rpm {**-V|--verify**} [**select-options**] [**verify-options**]

Verifying a package compares information about the installed files in the package with information about the files taken from the package metadata stored in the rpm database. Among other things, verifying compares the size, digest, permissions, type, owner and group of each file. Any discrepancies are displayed. Files that were not installed from the package, for example, documentation files excluded on installation using the "**--excludedocs**" option, will be silently ignored.

The package selection options are the same as for package querying (including package manifest files as arguments). Other options unique to verify mode are:

--nodeps
Don't verify dependencies of packages.

--nodigest
Don't verify package or header digests when reading.

--nofiles

Don't verify any attributes of package files.

--noscripts

Don't execute the **%verifyscript** scriptlet (if any).

--nosignature

Don't verify package or header signatures when reading.

--nolinkto
--nofiledigest (formerly **--nomd5**)
--nosize
--nouser
--nogroup
--nomtime
--nomode
--nordev

Don't verify the corresponding file attribute.

The format of the output is a string of 8 characters, a possible attribute marker:

```
c %config configuration file.
d %doc documentation file.
g %ghost file (i.e. the file contents are not included in the package payload).
l %license license file.
r %readme readme file.
```

from the package header, followed by the file name. Each of the 8 characters denotes the result of a comparison of **attribute**(s) of the file to the value of those **attribute**(s) recorded in the database. A single "**.**" (period) means the test passed, while a single "**?**" (question mark) indicates the test could not be performed (e.g. file permissions prevent reading). Otherwise, the (mnemonically em**B**oldened) character denotes failure of the corresponding **--verify** test:

```
S file Size differs
M Mode differs (includes permissions and file type)
5 digest (formerly MD5 sum) differs
D Device major/minor number mismatch
L readlink(2) path mismatch
U User ownership differs
G Group ownership differs
T mTime differs
P caPabilities differ
```

Digital Signature and Digest Verification

The general forms of rpm digital signature commands are

rpm --import ***PUBKEY*** ...

rpm {**--checksig**} [**--nosignature**] [**--nodigest**] ***PACKAGE_FILE** ...*

The **--checksig** option checks all the digests and signatures contained in *PACKAGE_FILE* to ensure the integrity and origin of the package. Note that signatures are now verified whenever a package is read, and **--checksig** is useful to verify all of the digests and signatures associated with a package.

Digital signatures cannot be verified without a public key. An ASCII armored public key can be added to the **rpm** database using **--import**. An imported public key is carried in a header, and key ring management is performed exactly like package management. For example, all currently imported public keys can be displayed by:

rpm -qa gpg-pubkey*

Details about a specific public key, when imported, can be displayed by querying. Here's information about the Red Hat GPG/DSA key:

rpm -qi gpg-pubkey-db42a60e

Finally, public keys can be erased after importing just like packages. Here's how to remove the Red Hat GPG/DSA key

rpm -e gpg-pubkey-db42a60e

Signing a Package

rpm --addsign|--resign *PACKAGE_FILE* *...*

Both of the **--addsign** and **--resign** options generate and insert new signatures for each package *PACKAGE_FILE* given, replacing any existing signatures. There are two options for historical reasons, there is no difference in behavior currently.

Using Gpg to Sign Packages

In order to sign packages using GPG, **rpm** must be configured to run GPG and be able to find a key ring with the appropriate keys. By default, **rpm** uses the same conventions as GPG to find key rings, namely the **$GNUPGHOME** environment variable. If your key rings are not located where GPG expects them to be, you will need to configure the macro **%_gpg_path** to be the location of the GPG key rings to use.

For compatibility with older versions of GPG, PGP, and rpm, only V3 OpenPGP signature packets should be configured. Either DSA or RSA verification algorithms can be used, but DSA is preferred.

If you want to be able to sign packages you create yourself, you also need to create your own public and secret key pair (see the GPG manual). You will also need to configure the

rpm macros

%_signature
The signature type. Right now only gpg and pgp are supported.

%_gpg_name
The name of the "user" whose key you wish to use to sign your packages.

For example, to be able to use GPG to sign packages as the user *"John Doe <jdoe@foo.com>"* from the key rings located in */etc/rpm/.gpg* using the executable */usr/bin/gpg* you would include

```
%_signature gpg
%_gpg_path /etc/rpm/.gpg
%_gpg_name John Doe <jdoe@foo.com>
%__gpg /usr/bin/gpg
```

in a macro configuration file. Use */etc/rpm/macros* for per-system configuration and *~/.rpmmacros* for per-user configuration. Typically it's sufficient to set just %_gpg_name.

Rebuild Database Options

The general form of an rpm rebuild database command is

rpm {**--initdb|--rebuilddb**} [**-v**] [**--dbpath** ***DIRECTORY***] [**--root** ***DIRECTORY***]

Use **--initdb** to create a new database if one doesn't already exist (existing database is not overwritten), use **--rebuilddb** to rebuild the database indices from the installed package headers.

Miscellaneous Commands

rpm --showrc
shows the values **rpm** will use for all of the options are currently set in *rpmrc* and *macros* configuration **file**(s).

rpm --setperms *PACKAGE_NAME*
sets permissions of files in the given package.

rpm --setugids *PACKAGE_NAME*
sets user/group ownership of files in the given package.

Ftp/Http Options

rpm can act as an FTP and/or HTTP client so that packages can be queried or installed from the internet. Package files for install, upgrade, and query operations may be specified as an **ftp** or ***http*** style URL:

ftp://USER:PASSWORD@HOST:PORT/path/to/package.rpm

If the **:PASSWORD** portion is omitted, the password will be prompted for (once per user/hostname pair). If both the user and password are omitted, anonymous **ftp** is used. In all cases, passive (PASV) **ftp** transfers are performed.

rpm allows the following options to be used with ftp URLs:

--ftpproxy *HOST*

The host *HOST* will be used as a proxy server for all ftp transfers, which allows users to ftp through firewall machines which use proxy systems. This option may also be specified by configuring the macro **%_ftpproxy**.

--ftpport *PORT*

The TCP *PORT* number to use for the ftp connection on the proxy ftp server instead of the default port. This option may also be specified by configuring the macro **%_ftpport**.

rpm allows the following options to be used with ***http*** URLs:

--*httpproxy HOST*

The host *HOST* will be used as a proxy server for all ***http*** transfers. This option may also be specified by configuring the macro **%_*httpproxy***.

--*httpport PORT*

The TCP *PORT* number to use for the ***http*** connection on the proxy http server instead of the default port. This option may also be specified by configuring the macro **%_*httpport***.

Legacy Issues

Executing rpmbuild

The build modes of rpm are now resident in the */usr/bin/rpmbuild* executable. Install the package containing **rpmbuild** (usually **rpm-build**) and see ***rpmbuild***(*8)* for documentation of all the **rpm** build modes.

Files

rpmrc Configuration

```
/usr/lib/rpm/rpmrc
/usr/lib/rpm/redhat/rpmrc
/etc/rpmrc
~/.rpmrc
```

Macro Configuration

```
/usr/lib/rpm/macros
/usr/lib/rpm/redhat/macros
/etc/rpm/macros
~/.rpmmacros
```

Database

```
/var/lib/rpm/Basenames
/var/lib/rpm/Conflictname
/var/lib/rpm/Dirnames
/var/lib/rpm/Filemd5s
/var/lib/rpm/Group
/var/lib/rpm/Installtid
/var/lib/rpm/Name
/var/lib/rpm/Packages
/var/lib/rpm/Providename
/var/lib/rpm/Provideversion
/var/lib/rpm/Pubkeys
/var/lib/rpm/Removed
/var/lib/rpm/Requirename
/var/lib/rpm/Requireversion
/var/lib/rpm/Sha1header
/var/lib/rpm/Sigmd5
/var/lib/rpm/Triggername
```

Temporary

*/var/tmp/rpm**

See Also

popt(3),
rpm2cpio(8),
rpmbuild(8),

rpm --help - as rpm supports customizing the options via popt aliases it's impossible to guarantee that what's described in the manual matches what's available.

http://www.rpm.org/ **<URL:http://www.rpm.org/>**

Authors

```
Marc Ewing <marc@redhat.com>
Jeff Johnson <jbj@redhat.com>
Erik Troan <ewt@redhat.com>
```

Referenced By

applydeltarpm(8), **apt**(8), **apt.conf**(5), **autoupdate**(8), **cmake28**(1), **cmake28-gui**(1), **cmake28modules**(1), **compat_digest.plug**(1), **cpack28**(1), **demo.plug**(1), **demofiles.plug**(1), **febootstrap**(8), **file2pacdep.plug**(1), **nodoc.plug**(1), **rpm2paco**(8), **rpm4**(3), **rpm_selinux**(8), **rpmcache**(8), **rpmconf**(8), **rpmdeps**(8), **rpmfile**(1), **rpmpeek**(1), **rpmreaper**(1), **rpmrebuild**(1), **rpmrebuild_plugins**(1), **set_tag.plug**(1), **superpaco**(8), **un_prelink.plug**(1), **uniq.plug**(1), **unset_tag.plug**(1), **yum-versionlock**(1)