# RPM RPM package tool

## CONCEPTS

rpm handles single package files and the local database. It does not download or resolve dependencies: use dnf for that.

**.rpm file** rpm -qp reads it
↓ rpm -U
**RPM database** rpm -q reads it
↓ rpm -V
**Files on disk** checked against db

Packages are named name-version-release.arch, e.g. tree-2.2.1-4.fc44.x86_64. Any leading part works as a query: tree or tree-2.2.1.

Add -p to any query to read a .rpm file instead of the installed database.

## QUERY INSTALLED

**rpm -qa**
List all installed packages

**rpm -qa '*<pattern>*'**
List packages matching a glob

**rpm -q *<pkg>***
Is it installed, and which version

**rpm -qi *<pkg>***
Show package details

**rpm -ql *<pkg>***
List files of a package

**rpm -qf *<path>***
Which package owns a file

**rpm -qc *<pkg>***
List its config files

**rpm -qd *<pkg>***
List its documentation files

**rpm -qa --last**
Packages by install time, newest first

**rpm -q --changelog *<pkg>***
Show the package changelog

**rpm -q --scripts *<pkg>***
Show install and remove scriptlets

## DEPENDENCIES

**rpm -qR *<pkg>***
What a package requires

## DEPENDENCIES (cont.)

**rpm -q --provides *<pkg>***
What a package provides

**rpm -q --whatprovides *<cap>***
Which package provides a capability

**rpm -q --whatrequires *<cap>***
Which installed packages need it

## PACKAGE FILES

**rpm -qip *<file.rpm>***
Details of a package file

**rpm -qlp *<file.rpm>***
List files in a package file

**rpm -qRp *<file.rpm>***
Requirements of a package file

**rpm -qp --scripts *<file.rpm>***
Scriptlets it would run

**rpm2cpio *<file.rpm>* | cpio -idmv**
Extract files without installing

## INSTALL, UPGRADE, REMOVE

**sudo rpm -Uvh *<file.rpm>***
Install or upgrade, with progress

**sudo rpm -ivh *<file.rpm>***
Install, keep other versions

**sudo rpm -Fvh *<file.rpm>***
Upgrade only if already installed

**sudo rpm -e *<pkg>***
Remove a package

**sudo rpm -Uvh --test *<file.rpm>***
Dry run: would it install?

**sudo rpm --reinstall *<file.rpm>***
Reinstall the same version

**sudo rpm -Uvh --oldpackage *<file.rpm>***
Downgrade to an older version

Prefer sudo dnf install ./<file.rpm>: it installs the local file and pulls in its dependencies.

## VERIFY

**rpm -V *<pkg>***
Check files against the database

**rpm -Va**
Verify every installed package

## VERIFY (cont.)

**rpm -Vf *<path>***
Verify the package owning a file

**sudo rpm --restore *<pkg>***
Reset owner, mode, mtime and caps

**rpm -V flags, e.g. S.5....T.**

| Char | Changed |
|---|---|
| S | size |
| M | mode or file type |
| 5 | digest (contents) |
| D | device number |
| L | symlink target |
| U / G | owner / group |
| T | mtime |
| P | capabilities |
| . / ? | passed / could not test |
| c d g l r | config, doc, ghost, license, readme file |

## SIGNATURES AND KEYS

**sudo rpmkeys --import *<key.asc>***
Trust a signing key

**rpmkeys -K *<file.rpm>***
Check digests and signature

**rpmkeys -Kv *<file.rpm>***
Show each signature and digest

**rpmkeys --list**
List imported keys

## QUERY FORMAT

**rpm -q --qf '%{NAME} %{VERSION}\n' *<pkg>***
Print chosen tags

## DATABASE AND CONFIG

**sudo rpmdb --rebuilddb**
Rebuild the package database

## RECIPES

### Which package owns this command

```
rpm -qf "$(command -v <cmd>)"
# not installed yet? ask dnf
dnf provides '*/bin/<cmd>'
```

### Look inside a download before installing

```
rpmkeys -K <file.rpm>
rpm -qip <file.rpm>
rpm -qp --scripts <file.rpm>
mkdir peek && cd peek &&
rpm2cpio ../<file.rpm> | cpio -idmv
```