Package management

Navid Ghassemi

What are Linux packages?

In Linux distributions, a “package” refers to a compressed file archive containing all of the files that come with a particular application. The files are usually stored in the package according to their relative installation paths on your system. Most packages also contain installation instructions for the OS, as well as a list of any other packages that are dependencies (prerequisites required for installation.)

Common types of Linux packages include .deb, .rpm, and .tgz.

Why package managers?

Since Linux packages do not usually contain the dependencies necessary to install them, many Linux distributions use package managers that automatically read dependencies files and download the packages needed before proceeding with the installation. Some examples of package managers are APT, YUM, and Pacman.

dpkg and apt

dpkg(debian package) is the software at the base of the package management system Debian and its numerous derivatives(like ubuntu). dpkg is used to install, remove, and provide information about .deb packages.

dpkg (Debian Package) itself is a low level tool. APT (Advanced Packaging Tool), a higher level tool, is more commonly used than dpkg as it can fetch packages from remote locations and deal with complex package relations, such as dependency resolution. Front ends for APT like aptitude (ncurses) are used for their friendlier interfaces.

How apt works

When you use apt to install a package, internally it uses dpkg. When you install a package using apt, it first creates a list of all the dependencies and downloads it from the repository.

Once the download is finished it calls dpkg to install all those files, satisfying all the dependencies.

Installing packages

As a toy example, we are going to install package htop, using apt-get

```
navidbeta@navidbetapc:~$ sudo apt-get install htop
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  htop
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 0 B/80.0 kB of archives.
After this operation, 221 kB of additional disk space will be used.
Selecting previously unselected package htop.
(Reading database ... 564325 files and directories currently installed.)
Preparing to unpack .../htop_2.1.0-3_amd64.deb ...
Unpacking htop (2.1.0-3) ...
Processing triggers for mime-support (3.60ubuntu1) ...
Processing triggers for desktop-file-utils (0.23-1ubuntu3) ...
Processing triggers for bamfdaemon (0.5.3+18.04.20180207.2-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for man-db (2.8.3-2) ...
Processing triggers for gnome-menus (3.13.3-11ubuntu1) ...
Setting up htop (2.1.0-3) ...
```

Installing from .deb file

You can install it using `sudo dpkg -i /path/to/deb/file` followed by `sudo apt-get install -f`.

You can install it using `sudo apt install ./name.deb` (or `/path/to/package/name.deb`). With old `apt-get` versions you must first move your deb file to `/var/cache/apt/archives/`directory. For both, after executing this command, it will automatically download its dependencies.

Installing from .deb file

Why to use `sudo apt-get install -f` after `sudo dpkg -i /path/to/deb/file` (mentioned in first method).

From `man apt-get`

`-f, --fix-broken Fix; attempt to correct a system with broken dependencies in place.`

When `dpkg` install a package and package dependency is not satisfied, it leaves the package in `unconfigured` state and that package is considered as broken.

`sudo apt-get install -f` command tries to fix this broken package by installing the missing dependency.

Repository

A repository is a collection of software for a linux distribution on a server. You grab information on the software that is available on the server using the packaging tools and download the software directly from those servers. As there are many folders in the web, they need to be kept separate. Ubuntu 5.10 will have different repositories than Ubuntu 5.04 or Mandriva or Redhat or Debian distros. Keeping the repositories split from each other ensures that your system will stay safe and not break due to incompatible software.

Package location

where these packages come from? from a Repository of different Repositories which are defined at `/etc/apt/sources.list` file and files located at `/etc/apt/sources.list.d/`.

To add a repository manually, use `add-apt-repository`.

Examples:

```
add-apt-repository 'deb http://packages.linuxmint.com/ julia main'
```

```
add-apt-repository ppa:gnome-desktop
```

PPA repository

Repositories are a more trustworthy way to download software. Since everything in the default repositories is reviewed by the Ubuntu team before it goes out, you know everything there is completely safe for your system. That doesn’t mean there aren’t downsides, though. Users typically have to wait for a new version of Ubuntu to try out new software. So if Firefox updates, you might not get to play with the new version until the next Ubuntu release comes out.

This is where PPAs come in. A PPA, or Personal Package Archive, is a collection of software not included in Ubuntu by default. Typically these repositories focus on a single program, but they can include more depending on the person maintaining them. A PPA might focus on an unreleased piece of software, such as Hotot, the best Linux Twitter client out there. It also might include updates for software already in Ubuntu, such as Firefox.

Removing packages

`apt-get remove htop`

Use `--purge` to remove the package and all of its configurations.

and if you want to remove automatically installed dependencies:

`$ apt-get autoremove bzr`

or even

`apt-get autoremove`

Searching for packages

```
apt-cache search htop
```

Updating

for updating a single package :

```
apt-get install htop
```

and for upgrading whatever installed:

```
apt-get upgrade
```

Reconfiguring packages

A program called debconf configures packages after they are installed. you can reconfigure a package (say htop) using

```
dpkg-reconfigure htop
```

Package information with dpkg

If you want to see what is inside a .deb file:

```
dpkg --contents bzr_2.6.0+bzr6595-1ubuntu1_all.deb

drwxr-xr-x root/root 0 2014-05-02 18:16 ./

drwxr-xr-x root/root 0 2014-05-02 18:15 ./etc/

drwxr-xr-x root/root 0 2014-05-02 18:15 ./etc/bash_completion.d/

-rw-r--r-- root/root 1467 2010-04-22 11:31 ./etc/bash_completion.d/bzr
```

dpkg -s

```
navidbeta@navidbetapc:~$ dpkg -s htop
Package: htop
Status: install ok installed
Priority: optional
Section: utils
Installed-Size: 216
Maintainer: Ubuntu Developers <ubuntu-devel-discuss@lists.ubuntu.com>
Architecture: amd64
Version: 2.1.0-3
Depends: libc6 (>= 2.15), libncursesw5 (>= 6), libtinfo5 (>= 6)
Suggests: lsof, strace
Description: interactive processes viewer
 Htop is an ncursed-based process viewer similar to top, but it
 allows one to scroll the list vertically and horizontally to see
 all processes and their full command lines.
 .
 Tasks related to processes (killing, renicing) can be done without
 entering their PIDs.
Original-Maintainer: Daniel Lange <DLange@debian.org>
Homepage: https://hisham.hm/htop/
```

dpkg -L

```
navidbeta@navidbetapc:~$ dpkg -L htop
/.
/usr
/usr/bin
/usr/bin/htop
/usr/share
/usr/share/applications
/usr/share/applications/htop.desktop
/usr/share/doc
/usr/share/doc/htop
/usr/share/doc/htop/AUTHORS
/usr/share/doc/htop/README
/usr/share/doc/htop/changelog.Debian.gz
/usr/share/doc/htop/copyright
/usr/share/man
/usr/share/man/man1
/usr/share/man/man1/htop.1.gz
/usr/share/pixmaps
/usr/share/pixmaps/htop.png
```

dpkg -S

```
navidbeta@navidbetapc:~$ dpkg -S /usr/bin/htop
htop: /usr/bin/htop
```

Thank you.