Package Management

APT, Dpkg & PPA

October 18, 2017

Table of Contents I

Table of Contents II

Why do I need a Package Management Tool?

Importance of Package Management for Administrators

- Manage software (install, update and remove)
- Close security gaps on your system
- Bug fixing
- Server distribution has no Graphical User Interface (GUI)
- Requiring very little user interaction
- Comfortable installation of specific tasks (group of applications) like servers
- Administrators have access to a huge pool of free applications and their dependencies

Software Package

- Software package is software that has been built form source with on eof the available package management systems.
- The package is typically provided as compiled code, with additional meta-information such as:
  - Software name
  - Description of its purpose
  - Version number
  - Vendor
  - Checksum
  - list of dependencies necessary for the software to run properly

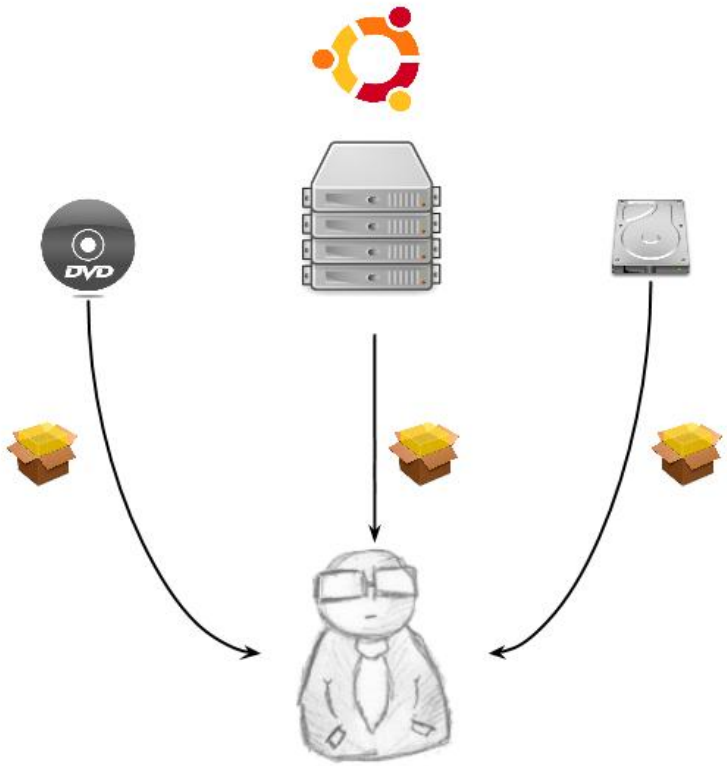
Package Management System

Package Management System is a collection of software tools that **automates** the process of **installing, upgrading, configuring, and removing** software packages for a computer's operating system.

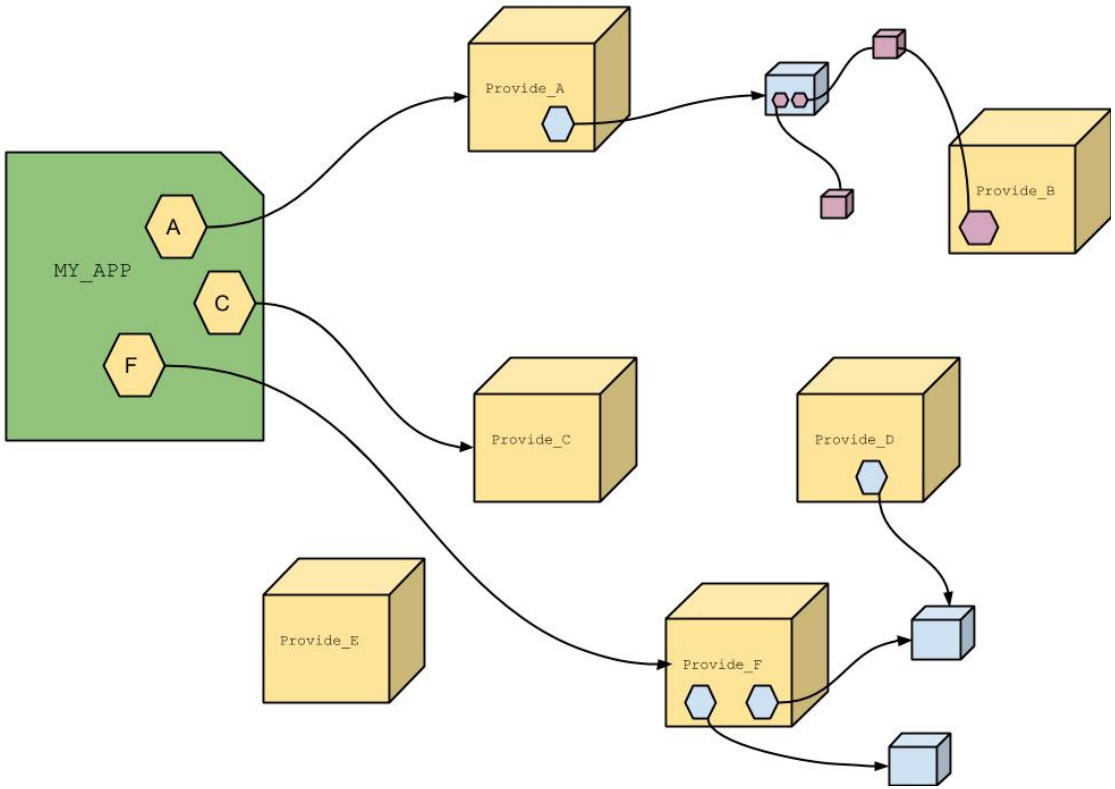
Software Repository

A software repository is a **storage location** from which software packages may be **retrieved** and **installed** on a computer.

Software & Repository



Package Dependencies



Advanced Package Tool

The Advanced Package Tool is a software that handle the installation and removal of software on Debian and other Linux distributions.

`APT` simplifies the process of managing software on Linux system.

Dpgk

Debian Package (dpkg) is the software for package management system in Debian and its derivates.

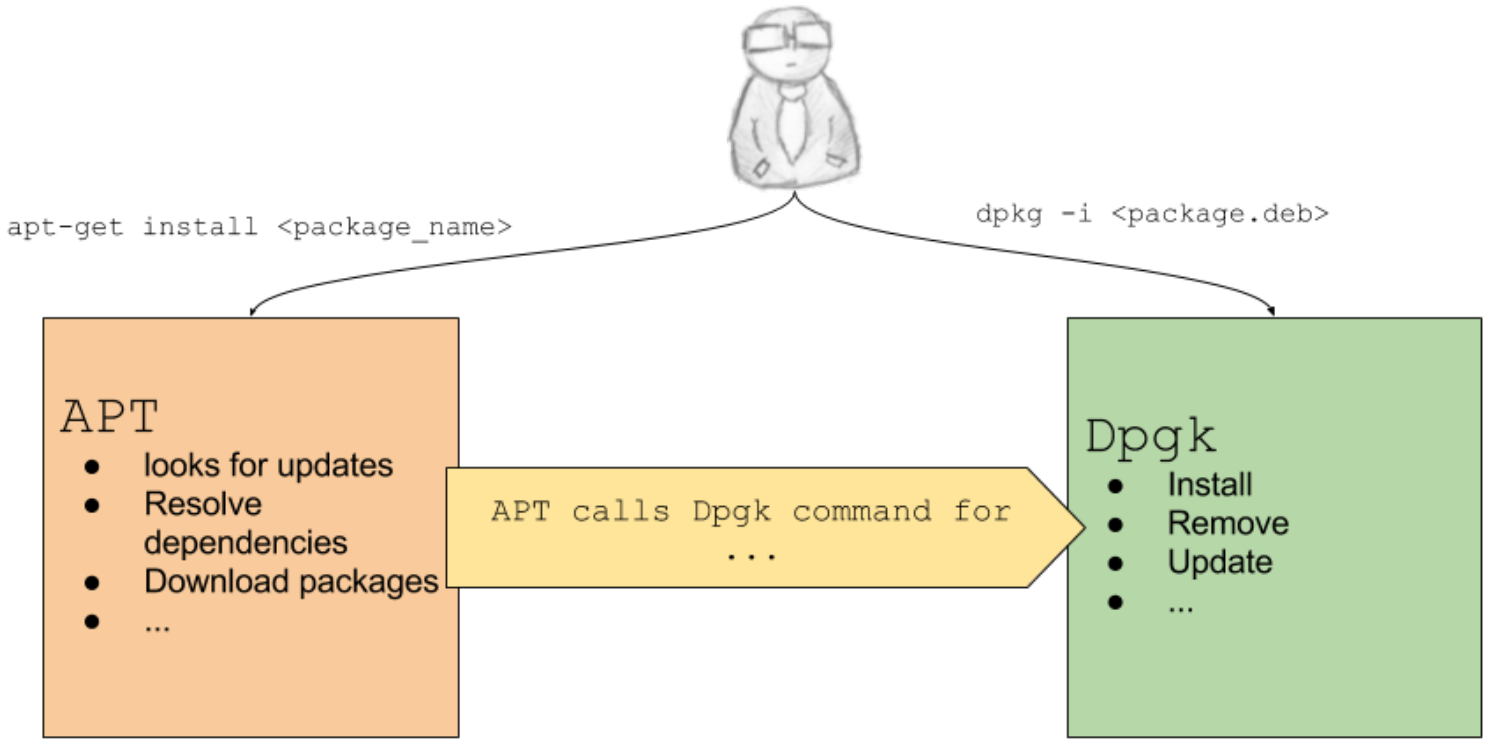
dpkg is used to install, remove and provide about .deb packages

dpkg is not sophisticated like APT.

Differences Dpkg & APT

Dpkg	APT
command only installs a package `[sudo] dpkg -i packageName.deb`	Package Mangement System `[sudo] apt-get install packageName`
It will notify the user of any dependencies that need to be installed, but it will not install them.	It will look for the dependencies and install them.
It does not have access to repositories to pull the dependencies from.	It is a set of tools to install, remove change packages.
You have to take care of the package order.	It determines packages that are not being used by any other, and will inform you that you can remove them.

APT & Dpkg


apt-get install <package_name>
dpkg -i <package.deb>
APT
looks for updates
Resolve dependencies
Download packages
...
APT calls Dpgk command for ...
Dpgk
Install
Remove
Update
...

File /etc/apt/sources.list I

```
deb http://de.archive.ubuntu.com/ubuntu trusty main restricted
```

- Type
  # deb or deb-src
- Source
  # http://de.archive.ubuntu.com/ubuntu
- Distribution
  # trusty, trusty-updates, trusty-security etc.
- Componentes
  # (optional): main restricted, universe, multiverse

File /etc/apt/sources.list II

- `[sudo] cp /etc/apt/sources.list /etc/apt/sources.list.bak`
  # Create a backup of your sources.list file before changing
- `[sudo] apt-get update`
  # Update apt after changing the file.

Repository Components

- `main`
  # **Canonical** maintained and only **free** software
- `restricted`
  # **Canonical** maintained and only **unfree** software
- `universe`
  # **Community** maintained and only **free** software
- `multiverse`
  # **Community** maintained and only **unfree** software

APT Commands and Methods

Command

- apt-cache
- apt-get

Method

- apt-pinning

APT-Cache I

- apt gathers information from the different sources listed in the `/etc/apt/sources.list` file.
- It does not manipulate the state of the system.
- Provides operations to search and show interesting package metadata.
- Stored in APT's internal database (Cache).
- This happen during the `apt-get update` operation.

APT-Cache II

`apt-cache [option(s)] commando [package]`

COMMAND	
search	Performs a full text search on all available package lists for the regex pattern given.
show	Displays the package records for the named packages.
pkgnames	Prints the name of each package in the system.
depends	Shows a listing of each dependency a package has and all the possible other packages that can fulfill that dependency.
policy	This is meant to help debug issues relating to the preferences file. With no arguments it will print out the priorities of each source.

APT-Get I

`[sudo] apt-get [option] commando [package_1] [package_N-1]`

COMMAND	
update	Update is used to resynchronize the package index files from their sources.
upgrade	Upgrade is used to install the newest versions of all packages currently installed on the system from the sources enumerated in **`/etc/apt/sources.list`**.
dist-upgrade	The function of upgrade and it handles changing dependencies with new version of packages. Pull in new packages that are not yet installed but are now required (often as part of upgrading to a new release).
install	Install is followed by one or more packages desired for installation or upgrading. Each package is a package name, not a fully qualified filename.
remove	Remove is identical to install except that packages are removed instead of installed.
purge	Purge is identical to remove except that packages are removed and purged. Additionally all config files of package
clean	Clean clears out the local repository of retrieved package files.

APT-Get II

```
[sudo] apt-get [option] command [package_1] [package_N-1]
```

OPTION	
-y, --yes	Assume “yes” as answer to all prompts and run non-interactively.
--purge	Use purge instead of remove for anything that would be removed. All config files will be deleted. `apt-get remove --purge <package_name>`
-d --download-only	Package files are only retrieved, not unpacked or installed.
--tar-only	Download only the tar file of a source archive.

APT-Pinning

Pinning controls which sources should be preferred.

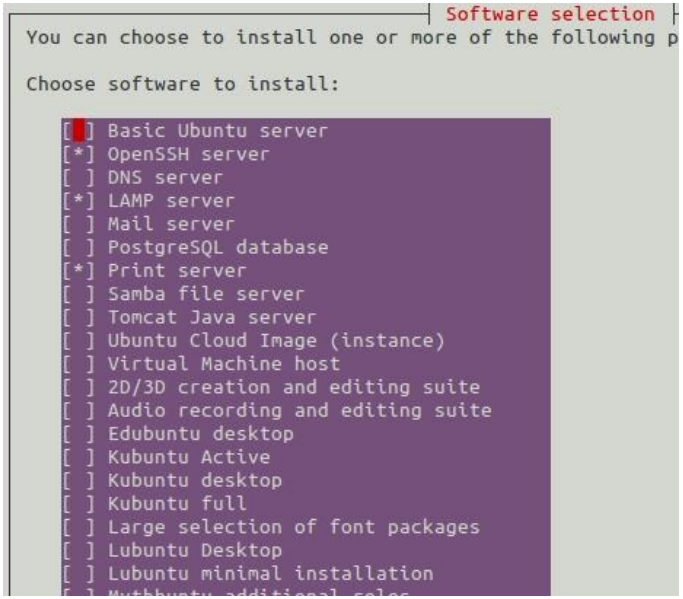
It forces APT to choose particular versions of packages which may be available in different versions from different repositories.

Scenarios

- Pinning per release
  - Only packages from newer or older version of Ubuntu
- Mix-and-match between stable, testing and unstable sources
  - For non-critical services like window manager
- Hold updates
  - Prevent updates to newest versions

Tasksel

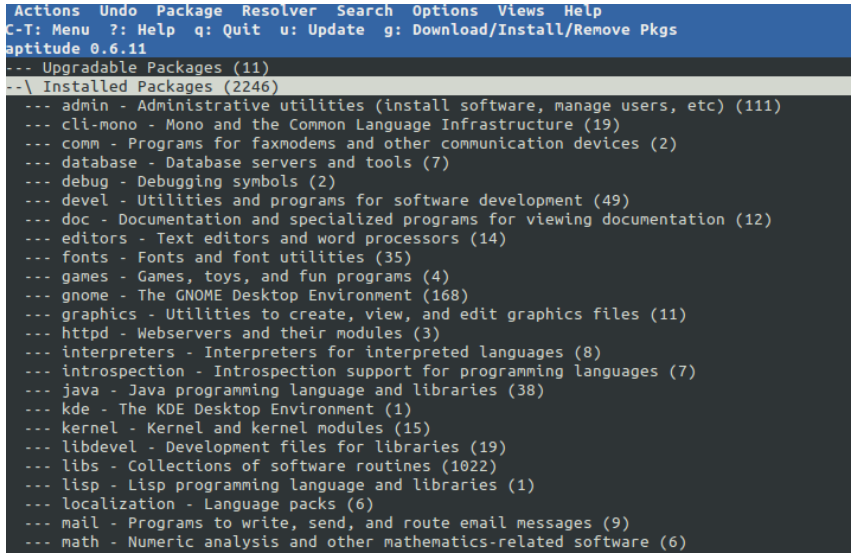
- Tasksel is a Debian/Ubuntu tool that installs multiple related packages as a coordinated "task" onto your system.
- `[sudo] tasksel --list-tasks`
- `[sudo] tasksel install lamp-server`



Tasksel Interface

Aptitude

- A **front-end** to the Advanced Packaging Tool.
- It **displays a list of packages** and allows the user to interactively **pick packages to install or remove**.
- It has an very **powerful search system** utilizing flexible search patterns.



Aptitude Interface

Personal Package Archive I

PPA allows you to upload and distribute Ubuntu source packages.

Use websites such as `www.launchpad.net` to host them in your own apt repository.

Software repository with latest software version between distro releases

- e.g. newest Nvidia driver or chromium beta

Personal Package Archive II

PPAs are provided by the community

- Anyone can create PPAs! Be aware of possible risks such as security and quality.
- Different status versions like beta, alpha or versions which don't work at all yet could be provided.
- Software installation at your own risk!

APT-Cacher I

Cache proxy for repository

One Server downloads the packages from the official server

It provides all other machines the packages to your LAN

Install packages from a local machine, instead of installing packages from the internet

APT-Cacher II