# Package Management

## A deeper look

Yunchih Chen
WSLAB
May 8, 2017

# Overview

- Motivation
- Package manager
- Various roles in package management: developer, maintainer, tester
- Quick overview of Debian
- Package life-cycle in the RedHat family, i.e. Fedora, RHEL, CentOS
- Package security

# Motivation

# Manaul Installation

Installation wizards like these are not **scalable**:


# Manual Installation

```
wget https://iperf.fr/download/source/iperf-3.1.3-source.tar.gz
tar zxf iperf-3.1.3-source.tar.gz
cd iperf && ./configure && make && sudo make install
```

```
git clone https://github.com/django/django.git
cd django
sudo python setup.py install
```

1. What if they have dependencies?
2. What if someday you want to remove them safely?
3. What if they conflict with installed files?
4. What if you can't afford compiling them?
5. What if the install scripts are **malicious**?
6. What if you want to upgrade them?

# Installing a webserver on Ubuntu in a breeze

```
root@ubuntu:~# apt-get install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0 ssl-cert
Suggested packages:
  www-browser apache2-doc apache2-suexec-pristine | apache2-suexec-custom
  openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 104 not upgraded.
Need to get 1,554 kB of archives.
After this operation, 6,412 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

# Oops ... I just ran Bumblebee's install script with root

install script does rm -rf /usr for ubuntu #123

Closed **ginoputrino** opened this issue on May 24, 2011 · 20 comments

**ginoputrino** commented on May 24, 2011

An extra space at line 351:
rm -rf /usr /lib/nvidia-current/xorg/xorg

causes the install.sh script to do an rm -rf on the /usr directory for people installing in ubuntu.

Totally uncool dude!!! The script deletes everything under /usr. I just had to reinstall linux on my pc to recover.

Removing the space will fix this. Probably should do it quickly!!!

1 7 9 2 33

https://github.com/MrMEEE/bumblebee-Old-and-abbandoned/issues/123

# Oops ... I just ran Steam's install script with root

**TcM1911** commented on Jan 16, 2015

pythoneer,

I believe the issue starts on line 19:

```
# figure out the absolute path to the script being run a bit
# non-obvious, the ${0%/*} pulls the path out of $0, cd's into the
# specified directory, then uses $PWD to figure out where that
# directory lives - and all this in a subshell, so we don't affect
# $PWD

STEAMROOT="$(cd "${0%/*}" && echo $PWD)"
STEAMDATA="$STEAMROOT"
```

This probably returns as empty which mean: `rm -rf "$STEAMROOT/"*` is the same ass `rm -rf "/"*` .

2 2

https://github.com/valvesoftware/steam-for-linux/issues/3671

# Quality Assurance

Packages are repeatedly tested before every release.

**Exercise**:

Briefly describe Fedora's testing plan before each release.
(Hint: google *Fedora release validation*)

# Package Manager

# Package Manager, heart of every Linux distribution

**dpkg / apt**


**pacman**


**rpm / yum / dnf**


**rpm / zypper**

# The goal of package manager

Enable the user to do the following things with ease:

- Search & install new software
- Upgrade software
- Safely remove software
- Verify the downloaded software content

# The goal of package manager

Enable the user to do the following things with ease:

- Search & install new software
  - Search package list in local database
  - Check conflict
  - Traverse dependency tree (*NP-complete* !)
- Upgrade software
  - Remove old version then install new version
- Safely remove software
- Verify the downloaded software content

# People

## Developer

Working on **upstream** project

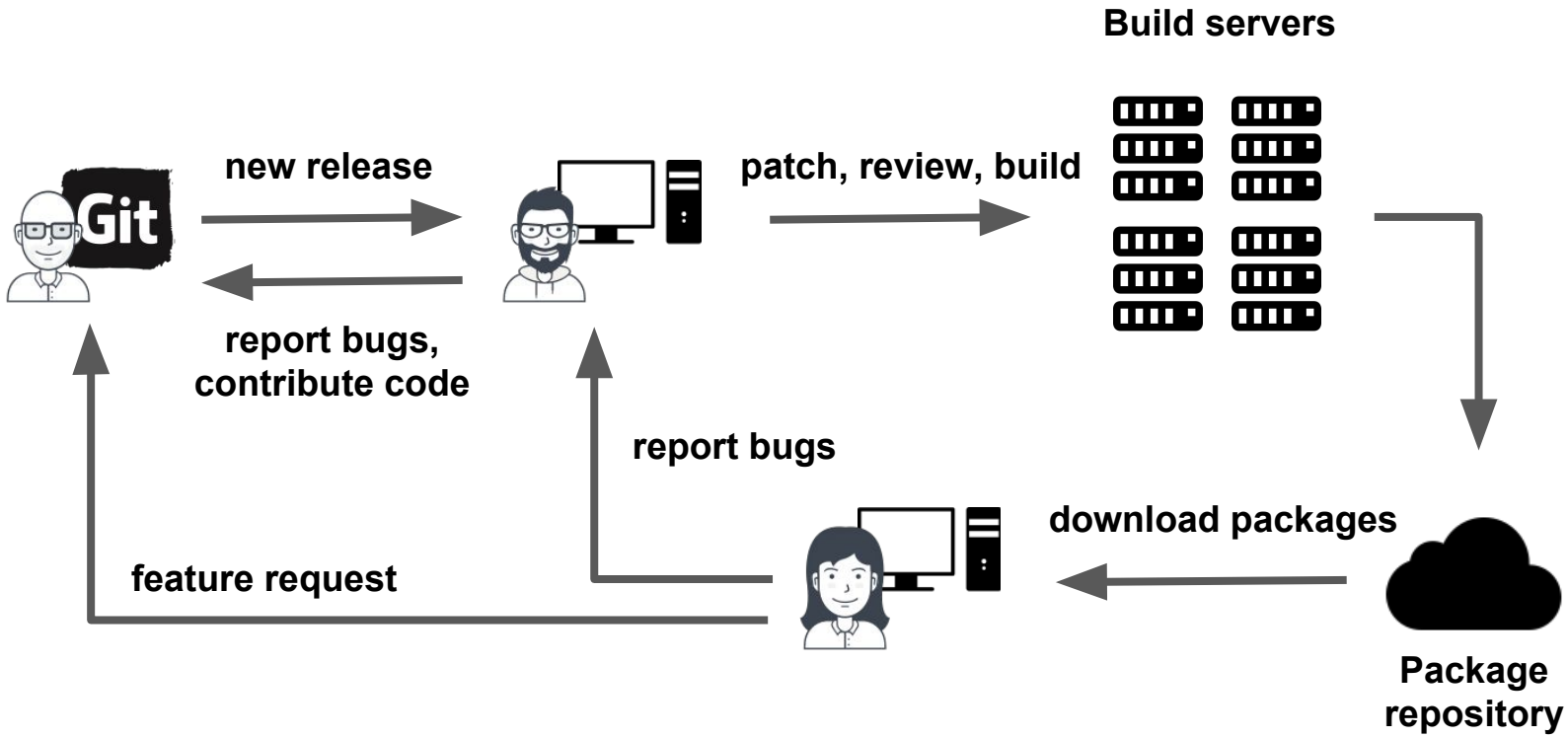
## Maintainer

* Every distribution has their own maintainers
* Create the distribution-specific experience
* Package stability, default options, usability

## User

You
Enjoy & give feedback

# Workflow


Build servers
Git
new release
patch, review, build
report bugs,
contribute code
report bugs
download packages
feature request
Package
repository

# Mirror

**Exercise**: What is the organization who hosts the primiary Debian mirror in Taiwan? (Hint: `ftp.tw.debian.org`)

- **Fun mirror**: `mirror.facebook.net`
- **Fast mirror**: `ftp.twaren.net`

# Vim as an example

## Vim experience on Ubuntu

```
vim-basic
vim-athena
vim-athena-py2
vim-gnome
vim-gnome-py2
vim-gtk
vim-gtk-py2
vim-gtk3
vim-gtk3-py2
vim-nox
vim-nox-py2
vim-scripts
vim-tiny
```

## Vim experience on Fedora

```
vim-x11
vim-minimal
vim-enhanced
```

- Different way of packaging
- Different default options
- Different plugin inclusion
- Different usability

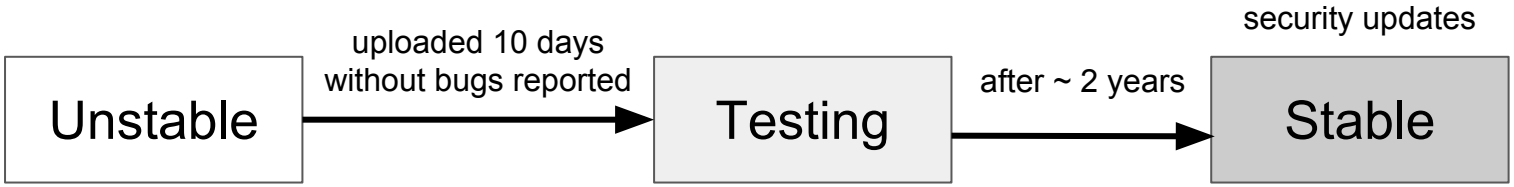
https://src.fedoraproject.org/cgit/rpms/vim.git/
https://packages.debian.org/sid/vim

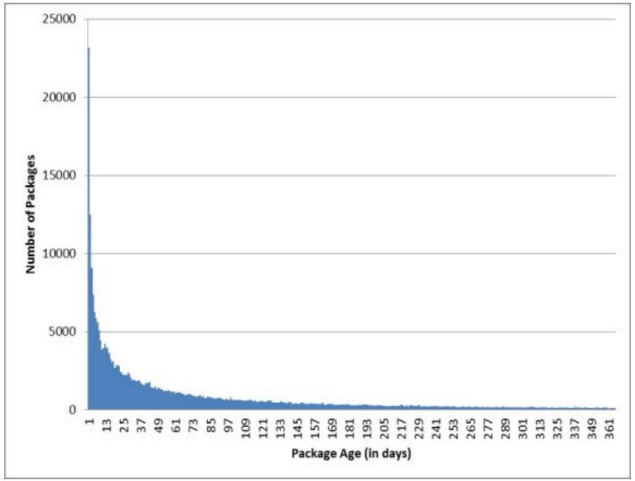
# Package Life-cycle

# Standard release v.s. Rolling release

- Standard release
  - Major package updates released in fixed cycle (six months for Fedora)
  - Packages well-tested when released
  - Only bugfix + small update between releases
  - **Long Term Support** (LTS)
  - Example: Ubuntu, Debian, Fedora
- Rolling release
  - No testing before shipping updates (Just Ship It!)
  - Good for the *adventurer*
  - Example: Arch Linux (CSIE workstation !!!!)

# Debian




https://debian-handbook.info/browse/stable/sect.release-lifecycle.html
Life and Death of Software Packages: An Evolutionary Study of Debian, CASCON '12 Proceedings of the 2012 Conference of the Center for Advanced Studies on Collaborative Research  Advanced Studies on Collaborative Research

# The Redhat family


* sponsored by Redhat
* Free
* 6 month release cycle
* Bleeding edge

* Long-term support
* Security fix
* Non-free

* Use RHEL codebase
* Free
* Not sponsored by Redhat

# Redhat, a giant in open source

The Linux Kernel


Gnome Desktop Environment


Xorg server, Systemd, NetworkManager


libvirt

# Fedora

- Include only **FREE** open source software.
  - Software must not be proprietary or patented
  - **Excluded**: MP3, Flash Player, Nvidia driver (not excluded in Ubuntu)
- The driving force of software innovation
  - NetworkManager, SELinux, Wayland, Systemd, etc.
- Six-month release cycle: reasonably stable new software

# Package Security

# Distribution keys

```
[root@ubuntu]# apt-key list
/etc/apt/trusted.gpg
--------------------
pub   1024D/437D05B5 2004-09-12
uid                  Ubuntu Archive Automatic Signing Key <ftpmaster@ubuntu.com>
sub   2048g/79164387 2004-09-12

pub   4096R/C0B21F32 2012-05-11
uid                  Ubuntu Archive Automatic Signing Key (2012) <ftpmaster@ubuntu.com>
......
```

```
[root@centos]# rpm -ql centos-release | grep KEY
/etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
/etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-Debug-7
/etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-Testing-7
[root@centos]# cat /etc/pki/rpm-gpg/RPM-GPG-KEY-CentOS-7
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.5 (GNU/Linux)
........
```

# Installing VirtualBox on CentOS

```
[root@centos]# cd /etc/yum.repos.d; wget http://download.virtualbox.org/virtualbox/rpm/rhel/virtualbox.repo
[root@centos]# yum --enablerepo=epel install dkms
Retrieving key from https://www.virtualbox.org/download/oracle_vbox.asc
Importing GPG key 0x98AB5139:
 Userid     : "Oracle Corporation (VirtualBox archive signing key) <info@virtualbox.org>"
 Fingerprint: 7b0f ab3a 13b9 0743 5925 d9c9 5442 2a4b 98ab 5139
 From       : https://www.virtualbox.org/download/oracle_vbox.asc
Is this ok [y/N]: y

....

[root@centos]# yum install VirtualBox-4.1
```

**Typing y means you trust the repository!**

# Distribution keys

- A set of public keys imported when you enable a repository
- When installing new packages, binary content checked against the keys
- Only the person who signs the package has the private key
- Prevent *Man-in-the-middle* attack
  - Attacker takes control of a package mirror
  - Add malicious code into package
  - Add malicious dependencies into package metadata
  - Download package via HTTP instead of HTTPS

# Distribution keys (2)

- CentOS, Ubuntu store just a few keys, either in plain text or keyring
- Arch Linux stores many keys owned by core maintainers
  - `pacman-key -l`
- Language package repository like PyPI, Rubygem allows arbitrary developers to upload packages. **Hard to enforce package signing.**

🚫 **`sudo pip install xxx`**

- Further security enhancement: *Debian's Reproducible Builds*

See: https://wiki.debian.org/ReproducibleBuilds/About

# Recent example

HandBrake mirror hacked and Mac OS X users compromised:

https://www.cyberciti.biz/open-source/handbrake-for-mac-mirror-server-was-compromised-and-infected-with-proton-malware/

**Exercise**: read the article and describe how package manager can protect you from attack like this.