RPM and Yum

How to manage your packages without UPS

Nick Urbanik <nicku@nicku.org>

This document Licensed under GPL—see section 9

Outline

Contents

1 What is RPM?

What is RPM?

- RPM stands for The RPM Package Manager.
  - It used to stand for Red Hat Package Manager
- It is used to manage *packages*.
  - "Oh, I *see*."
  - . . . no I don't. . .

1.1 What is a Package?

What is an RPM Package?

- It's a piece of software, all put in one RPM file
- For example, I am using the. . . let's see. . .
  `$ rpm -q evince ↩`
  `evince-0.5.1-3`
  . . . software package to display these notes.
- The *name* of the package is `evince`.
- The *version* of the package is 0.5.1
- The *release* is 3.

An RPM Package usually contains files

- My `evince` package has files inside:
  `$ rpm -ql evince ↩`
  ```
  /etc/gconf/schemas/evince-thumbnailer.schemas
  /etc/gconf/schemas/evince.schemas
  /usr/bin/evince
  ...
  /usr/share/omf/evince/evince-sr.omf
  /usr/share/omf/evince/evince-uk.omf
  ```

1.2 Why should I have packages? Why not just zip files or something?

Why should I have packages? Why not just zip files or something?

- We could just have zip files, or tar balls, or something like that, with just the files inside?
- It could work.
- So why have RPM packages?

1.3 Advantages of RPM packages

Advantages of RPM packages

- You can install a new package easily with one operation
- You can uninstall a package just as easily
- the package can contain information about what else it needs to work properly
- RPM won't clobber configuration files, and does sensible things with them
- The RPM system includes a database with lots of information about how the files should be installed, their size, owner, type,...

2 RPM Commands

RPM Commands

There are seven main RPM operations:

- For installing software packages:
  - *update*, *install*, *freshen*

- For making *queries*
- for *erasing* packages
- for *verifying* that a software package is correctly installed
- for *checking signatures* — verifying that an RPM package file is really from the people you trust, and not from Joe Trojan Cracker

2.1 Queries

Queries

- You saw above that I can list the files with
  `$ rpm -ql evince ↩`
- I can also see information about the package with
  `$ rpm -qi evince ↩`
- Try it now!

3 Mount the NFS disk

1. I have put some RPM packages on our NFS server, "`gw`"
2. I would like you to "mount" the directory `/inst_disk/ppracer` from that server
3. To do that:
   (a) Set up `sudo` (see my handout on `sudo`)
   (b) Create a directory to mount this directory on:
   `$ mkdir ~/mnt ↩`
   (c) Now mount the NFS directory on the directory you just made:
   `$ sudo mount gw:/inst_disk/ppracer ~/mnt ↩`
   (d) Now change to that directory and list the files:
   `$ cd ~/mnt ↩`
   `$ ls -l ↩`
   (e) What do you see?

3.1 Install

Install

- This will install a software package, regardless of whether there is a previous version installed
- This is always the right thing to do for kernels.

3.2 Installing a Kernel

1. See what kernels are installed:
   $ **rpm -q kernel** ↩
2. Change to where you mounted your NFS disk
3. list the files to see if you can find the kernel RPM
4. install it with:
   $ **sudo rpm -ihv kernel-2.6.16-1.2069_FC4.i686.rpm** ↩
5. See what kernels are installed again:
   $ **rpm -q kernel** ↩

3.3 Update

Update

- Install a package if it is not installed, or
- install a newer version if there is a newer version available…
- … otherwise, do nothing.
- Let's use this command to install the wonderful Planet Penguin Racer game:
- $ **cd ~/mnt** ↩
- $ **ls -l** ↩
- $ **sudo rpm -Uhv $(ls|grep -v kernel)** ↩

3.4 Freshen

Freshen

- Install a new version of a package only if an older one is already installed, otherwise do nothing.

3.5 Erasing

Erasing

- Remove a package:
  $ **sudo rpm -e ppracer** ↩

3.6 Verfying

Verfying

- Checking the files in a package all match the original data
- If it doesn't match, then flags show this:

flag	*what it means*
S	file *S*ize differs
M	*M*ode differs (includes permissions and file type)
5	MD*5* sum differs
D	*D*evice major/minor number mismatch
L	symbolic *L*ink points to wrong place
U	*U*ser ownership differs
G	*G*roup ownership differs
T	m*T*ime differs
C	selinux security *C*ontext differs

$ **$ rpm -V setup** ↩

```
  S.5....T. c /etc/aliases
S.5....TC c /etc/bashrc
........C c /etc/exports
S.5....T. c /etc/printcap
S.5....TC c /etc/profile
S.?....T. c /etc/securetty
```

3.7 Checking Digital Signatures

Checking Digital Signatures

- First, you need to *import* the public keys that are provided by Fedora:
  $ **sudo rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY*** ↩
- $ **rpm -K** ⟨*package-file.rpm*⟩ ↩

3.8 Exercise with signatures

1. Go to the NFS mounted directory:
   $ **cd ~/mnt** ↩
   … and check the file are there:
   $ **ls** ↩

2. Check the signatures on all the files:
   `$ rpm -K *.rpm` ↩
3. If you get messages about keys missing, then *import* them:
   `$ sudo rpm --import /etc/pki/rpm-gpg/RPM-GPG-KEY*` ↩
4. . . . then check the signatures again:
   `$ rpm -K *.rpm` ↩

4 RPM Command Examples

RPM Command Examples

command	effect
`rpm -qa | less`	list all installed software packages
`rpm -q httpd`	show the version of the httpd package, if it is installed
`rpm -qa | grep httpd`	show all installed packages that have *httpd* in their name
`rpm -ql httpd`	***l***ist all files in the httpd package
`rpm -qd httpd`	list all ***d***ocumentation files in the httpd package
`rpm -qc httpd`	list all ***c***onfiguration files in the httpd package
`rpm -qi httpd`	display ***i***nformation about the package
`rpm -V httpd`	***v***erify that the `httpd` package is correctly installed
`rpm -qf /etc/passwd`	determine which package the `/etc/passwd` ***f***ile belongs to

4.1 A quick comparison with `dpkg`

A quick comparison with `dpkg`

command	effect
`dpkg -list | less`	list all installed software packages
`dpkg -l httpd`	show the version of the `httpd` package, if it is installed
`dpkg -list | grep httpd`	show all installed packages that have *httpd* in their name
`dpkg -listfiles httpd`	list all files in the `httpd` package
`dpkg -print-avail httpd`	display information about the package
`dpkg -S /etc/passwd`	determine which package the `/etc/passwd` file belongs to

5 Yum — Sounds delicious!

Yum

- Yum is a high level wrapper for RPM
- Uses the information in the RPM packages to figure out what things to install
- downloads the files you need, and installs them, lovingly taking care of which packages depend on what.

5.1 Why not stick with RPM?

Why not stick with RPM?

- RPM is good, but it's too low level to easily manage all the packages on the system
- Yum uses the information in the RPM database, and the RPM files themselves to make sure that each software package has everything it needs.

5.2 What's all this primative text stuff?

What's all this primative text stuff?

- There are many GUI programs that use yum as their backend
- I have never used any of them, so your exercise is to Google for them!

6 Updates

Updates

- Software has bugs
- Fedora particularly has many new versions of software that are being made available, since it aims to provide the current version of most pieces of software.
- Fedora has more updates than any other stable distribution I know of.
- Only a few are for security reasons, most are for updates.

6.1 Why update? Why not update?

Why update? Why not update?

Why update?

- The easiest systems to crack have no updates.

Why not update?

- Updating can break a working system
- But that doesn't stop me from keeping my systems updated!
- ... but not good if your business loses thousands of dollars per second of downtime.
  - In such cases, you have a test system where you test the updates first, check they don't stop your applications from working

6.2 How to update

How to update

Do this once only:

- Ensure your own account is in the `sudoers` file
  - see the `sudo` handout
- set your `http_proxy` environment variable
  - See the next section: 6.3 on the following page

Do this for each update:

- then do:
  `$ sudo yum -y update ↩`
- and watch it go.

6.3 Setting `http_proxy`

Setting `http_proxy`

If access to web servers on the Internet is blocked by a firewall, and a proxy server is required, such as here in the TAFE:

- You need to set the `http_proxy` environment variable.
- The proxy server here has the DNS name "`gw`", and listens on port 5865 ...
- ... so set the `http_proxy` like this:
  `export http_proxy=http://gw:5865/`
- The best thing to do is to edit your login script `~/.bash_profile` and add it at the end:
  `emacs ~/.bash_profile &`
- Then log out and log in...
- You should be able to see the value now:
  `$ echo $http_proxy ↩`
  `http://gw:5865/`

7 The "repos": repositories

The "repos": repositories

- In the `/etc/yum.repos.d` directory are some files that end with "`.repo`"
- You can change these, add other repos (such as dag, freshrpms, atrpms, livna)
- You can also change the `baseurl` to a local Australian mirror (see `http://fedora.redhat.com/Download/mirrors.html` for a list of Australian mirrors)

7.1 Adding extras

Adding extras

- Well, the fedora extras repository is already enabled, so nothing to do here.

7.2 Digital signatures

- If you add the `dag`, `freshrpms`, `atrpms` or `livna` repositories, you should import their public keys, and enable checking the GPG signatures of the packages.
- For example, here is my `/etc/yum.repos.d/dag.repos` file:
```
[dag]
name=Dag RPM Repository for Fedora Core
baseurl=http://apt.sw.be/fedora/$releasever/en/$basearch/dag/
enabled=1
gpgcheck=1
```

8 Some Things to Read

References

[1] Dr. Peter Salus. *The Daemon, the GNU & the Penguin*. Grocklaw 2006. `http://www.groklaw.net/staticpages/index.php?page=20051013231901859`.

[2] Eric Foster-Johnson. *RPM Guide*. `http://fedora.redhat.com/docs/drafts/rpm-guide-en/`

[3] *The RPM Package Manager*. `http://rpm.org/`

[4] Wikipedia entry. *Yellow dog Updater, Modified*. `http://en.wikipedia.org/wiki/Yellow_dog_Updater,_Modified`

9 License Of This Document

License Of This Document
Copyright © 2006 Nick Urbanik <nicku@nicku.org>
You can redistribute modified or unmodified copies of this document provided that this copyright notice and this permission notice are preserved on all copies under the terms of the GNU General Public License as published by the Free Software Foundation—either version 2 of the License or (at your option) any later version.