RPM 101

The Red Hat Package Manager

An Introduction

Special Thanks

- Edward C.Bailey - Maximum RPM
- Dan Poirer - IBM
- Red Hat
- Others

RPM 101 Outline

- What is RPM and why should we use it
- Basic RPM functions
- Special Parameters or modifiers
- Directory layout and files
- Building packages and development
- The SPEC file

Why RPM ?

- RPM is a tool to manage applications and groups of applications.
- RPM packages can be verified prior to installation.
- RPM packages can be easily removed.
- Entire systems can be upgraded easily.
- RPM use can reduce the effort required to maintain and upgrade a system.

What is an RPM

- Binary packages - .rpm
  - Binary package
  - Platform specific
  - Precompiled and linked
- Source packages - .srpm
  - Source code - .src.rpm or .srpm
  - Not pre-compiled

Non-RPM application

- `cc -o foobar foobar.c`
- `./configure ;make ;make install`
- `foobar.conf` file editing
- Documentation installation
- Removal difficult
- A different method for different packages

Basic commands

- Install: `rpm -i foobar.rpm`
- Erase: `rpm -e foobar`
- Upgrade: `rpm -U foobar.rpm`
- Freshen: `rpm -f`
- Verify: `rpm -V foobar.rpm`
- Query: `rpm -q`

General options

Verbose	`-v`
Very verbose	`-vv`
Alternate rpm root :	`--root <path>`
Alternate rpmrc file:	`--rcfile <rcfile>`
Alternate database:	`--dbpath <path>`

RPM package install

- Download the package
  - binary - .rpm
  - source - .srpm or src.rpm
- Binary install: **`rpm -i foobar.rpm`**
- Source install: **`rpm -i foobar.srpm`**
  - Caution: Use a binary file download

Install options

Print hash marks:	`-h` (or `--hash`)
Installation tests only:	`--test`
Don’t check dependencies:	`--nodeps`
Ignore package & file conflicts:	`--force`
Replace files from another package:	`--replacefiles`
Ignore package architecture:	`--ignorearch`
Ignore package operating system:	`--ignoreos`

RPM -e foobar (Erase)

- Database checked for dependancies
- Pre-uninstall script executed
- Modified config files saved
- All package files deleted
- Post-uninstall script executed

Erase options

Perform erase tests only:	`--test`
Don’t execute pre/post erase scripts:	`--noscripts`
Do not check for dependencies:	`--nodeps`

Rpm -U foobar.rpm

- Package upgrade
- Old package is erased
- New package is installed
- All config files upgraded
- Old modified config files saved as **`file.rpmsave`**

Upgrade options

- Most options are the same as install.
- "Upgrade" to an older package: `--oldpackage`
  - Provides a "nice" way to back out when a upgraded package fails.
- Ignore file and package conflicts: `--force`
  - Make sure your backups are current !!!!

RPM package verification

- Verify package was correctly installed
  - Verify syntax: `rpm -V foobar.rpm`
- Verify package file has not been modified
  - `rpm -K foobar.rpm`
- PGP
  - Digital signature
  - PGP public and private keys

RPM Freshen

- `rpm -f foobar.rpm`
- `rpm -f *`
- A selective upgrade:
  - Upgrades only those packages currently installed

Information please

- `rpm -q` (or `--query`) options
  - Query selection by package
    - Package file: `-p <file>`
    - Package owning <file>: `-f <file>`
    - All packages: `-a`
    - group <group> `-g <group>`
    - Dependencies: `--whatrequires`

Query details

- Display the full package label
  - `rpm -q -p foobar-1.1.2ac <null>`
- Additional options
  - Summary: `-i`
  - List files in a package: `-l`(`lc L`)
  - List configuration files: `-c`
  - List documentation files `-d`
  - List files in a package w/state: `-s`

Miscellania

- Rebuild the rpm database: `--rebuilddb`
- Create new RPM database: `--initdb`
- Limit output: `--quiet`
- Help: `--help`
- Display rpm version: `--version`
- Extract file from package: `rpmcpio`

Files (RedHat)

- /etc/rpmrc
- /usr/src/redhat/SOURCES
- /usr/src/redhat/SPECS
- /usr/src/redhat/BUILD
- /usr/src/redhat/RPMS
- /usr/src/redhat/SRPMS

Files (SuSE)

- /usr/src/packages
- /usr/src/packages/BUILD
- /usr/src/packages/SOURCES
- /usr/src/packages/SPECS
- /usr/src/packages/RPMS
- /usr/src/packages/SRPMS

SPEC files

- Creating the Spec File
- The Preamble section
- The %prep Section
- The %build Section
- The %install Section
- The %files Section
- The install/uninstall scripts section
- The %clean section

Package build

- `rpm -ba foobar.SPECS`
- Results:
  - `foobar.rpm`
  - `foobar.srpm`

References

- Maximum RPM, Taking the Red Hat Package Manager to the Limit,
  - By Edward C. Bailey
- Don Poiner, Software Engineer, IBM
  - Packaging software with RPM, Part 1, 2 and 3
    - http://www-106.ibm.com/developerworks/library/l-rpm1/
    - http://www-106.ibm.com/developerworks/library/l-rpm2/
    - http://www-106.ibm.com/developerworks/linux/library/l-rpm3.html
- The great folks @ RPM.org, http://www.rpm.org

-
- Good luck and fortunes as you use RPM
  - Anonymous