

第5章 软件包管理

宣江华

Why RPM?

- 软件包如何:
 - 安装?
 - 升级?
 - 查询?
 - 校验?
 - 卸载?

what's RPM

- **RedHat Package Manager(软件包管理器)**
 - 开放性：任何人都可以免费使用
 - 通用性：很多Linux版本采用
 - 用户：方便管理软件包
 - 开发者：

1.1 RPM 的设计目标

- 1. 可升级性
 - 升级个别组件、全部组件
- 2. 强大的查询功能
 - 包信息、安装文件
- 3. 系统校验
- 4. 纯净源码
 - 有纯净源码、使用过的补丁、以及完整的建构指令。

1.2 RPM 基本操作模式

- RPM 有五种基本操作模式
 - 安装、卸载、升级、查询和校验
 - （不包括软件包建构）
 - rpm --help 命令。

1.2 RPM 基本操作模式

■ RPM包的名称格式

- **bc-1.06-21.i386.rpm**
- 该文件名包括软件包名称“bc”；
- 软件的版本“1.06”其中包括主版本号和次版本号；
- “i386”是软件所运行硬件平台；
- 最后“rpm”做为文件的扩展名，当然代表文件的类型为RPM包。

1.2.1 安装RPM包

- 命令格式:
- **rpm -ivh RPM包的全文件名**
- 在此命令的参数选项中:
 - **i**: 代表安装。
 - **v**: 代表**verbose**, 设置在安装过程中将显示详细的信息。
 - **h**: 代表**hash**, 设置在安装过程中将显示“#”来表示安装的进度。

1.2.2 删除 RPM包

- 命令格式：
 - **rpm -e RPM包名称**
- 注意：软件包的名称，而不是软件包安装文件名。
- 例如：**rpm -e bc**

1.2.2 删除 RPM包（举例）

- （1）首先在终端上将系统中自带的输入法删除。
 - `# rpm -e miniChinput`
- （2）进入到下载fcitx的目录（假设是目录）
 - `# cd /input`
- （3）然后使用RPM命令来安装fcitx
 - `# rpm -ivh fcitx-1.8.5-1.rpm`

1.2.3 升级RPM包

- 命令格式：
 - rpm -U RPM包的全路径文件名
- 例如：
 - rpm -U bc-1.2-1.i386.rpm

1.2.4 查询RPM包

- 命令格式: `rpm -q`
- 使用 `rpm -q` 命令来查询安装的软件包的数据库
- 例如: `rpm -q fcitx`命令会显示安装的软件包 `fcitx`的名称、版本、和发行号码。

1.2.4 查询RPM包（续）

- **-a:**
 - 查询所有已安装的软件包。
- **-f <file>:**
 - 会查询拥有 <file> 的软件包。
- **-p <packagefile>:**
 - 查询软件包 <packagefile>。
- **-i :**
 - 显示软件包信息。
- **-l:**
 - 显示软件包所含的文件列表。

1.2.4 查询RPM包（举例）

- 查询**cv**s软件包在系统中是否安装
 - `$ rpm -q cvs`
- 查询系统中已安装的**cv**s软件包的描述信息
 - `$ rpm -qi cvs`
- 查询系统中已安装的**cv**s软件包里所包含的文件
 - `$ rpm -ql cvs`

1.2.4 查询RPM包（举例）

- 查询系统中文件/etc/passwd所属的软件包
 - `$ rpm -qf /etc/passwd`
- 查询cvs-1.11.2-10.i386.rpm包文件中的信息
 - `$ rpm -qp cvs-1.11.2-10.i386.rpm`
- 查询系统中已经安装的所有包含名字php的软件包
 - `$ rpm -qa | grep php`

```
[root@localhost root]# rpm -ivh xmms-1.2.7-21.p.i386.rpm
warning: xmms-1.2.7-21.p.i386.rpm: V3 DSA signature: NOKEY, key ID db42a60e
Preparing... ##### [100%]
 1:xmms ##### [100%]
[root@localhost root]# rpm -q xmms
xmms-1.2.7-21.p
[root@localhost root]# rpm -qa | more
setup-2.5.25-1
bzip2-libs-1.0.2-8
e2fsprogs-1.32-6
glib-1.2.10-10
iputils-20020927-2
losetup-2.11y-9
net-tools-1.60-12
shadow-utils-4.0.3-6
libtermcap-2.0.8-35
MAKEDEV-3.3.2-5
raidtools-1.00.3-2
hotplug-2002_04_01-17
file-3.39-9
findutils-4.1.7-9
ash-0.3.8-8
modutils-2.4.22-8
```

```
[root@localhost root]# rpm -ql xnmms
/usr/bin/wmxnmms
/usr/bin/xnmms
/usr/lib/libxnmms.so.1
/usr/lib/libxnmms.so.1.2.1
/usr/lib/xnmms
/usr/lib/xnmms/Effect
/usr/lib/xnmms/Effect/libecho.so
/usr/lib/xnmms/Effect/libstereo.so
/usr/lib/xnmms/Effect/libvoice.so
/usr/lib/xnmms/General
/usr/lib/xnmms/General/libir.so
/usr/lib/xnmms/General/libjoy.so
/usr/lib/xnmms/General/libsong_change.so
/usr/lib/xnmms/Input
```

```
[root@localhost root]# rpm -qf /usr/lib/libxnmms.so.1
xnmms-1.2.7-21.p
[root@localhost root]# rpm -qa | grep xnmms
xnmms-1.2.7-21.p
[root@localhost root]#
```


1.2.5 校验RPM包

- 命令格式:
- **rpm -V**
- 校验软件包比较从某软件包安装的文件和原始软件包中的同一文件的信息。
- 校验的最简单用法是 **rpm -V fcitx**，它校验所有在 **fcitx**软件包内的文件是否和最初安装时一样。

1.2.5 校验RPM包（续）

- 要校验包含某一特定文件的软件包：
 - `# rpm -Vf /bin/vi`
- 要校验所有安装了的软件包：
 - `# rpm -Va`
- 要根据 **RPM** 软件包文件来校验安装了的软件包：
 - `# rpm -Vp fcitx-1.8.5-1.rpm`
 - 单用一个 .（点）意味着测试通过。下列字符代表某类测试失败：
 - **5** — **MD5** 校验和、**S** — 文件大小、**L** — 符号链接
 - **T** — 文件修改时间、**D** — 设备、**U** — 用户
 - **G** — 组群、**M** — 模式（包括权限和文件类型）
 - **?** — 不可读文件

```
[root@localhost root]# rpm -V xmms
[root@localhost root]# rpm -q xmms
xmms-1.2.7-21.p
[root@localhost root]# rpm -e xmms
[root@localhost root]# rpm -q xmms
package xmms is not installed
[root@localhost root]#
```

2 文件压缩打包（gzip）

- **Linux**下常用的压缩和解压缩命令。
- 由官方仓库的**gzip**软件包提供。
- 压缩后 **gzip** 会在**每个文件的后面添加扩展名 .gz**。
- 压缩后原文件会被**自动删除**。
- 在 **windows** 下可以用 **winzip** 或 **winrar**或 **7-zip** 解压。

2 文件压缩打包（gzip）

常用选项	说明
-c	将输出写到标准输出上，并保留原有文件。
-d	将压缩文件解压。
-l	对每个压缩文件，显示压缩文件的大小、未压缩文件的大小、压缩比、未压缩文件的名字等详细信息
-r	递归式地查找指定目录并压缩其中的所有文件或者是解压缩。
-t	测试，检查压缩文件是否完整。
-v	对每一个压缩和解压的文件，显示文件名和压缩比。

gzip命令举例

- 压缩文件filename
\$ gzip filename
- 压缩文件 file1和file2并显示执行过程
\$ gzip -v file1 file2
- 递归地高度压缩mydir目录下的所有文件（逐个文件进行）
\$ gzip -9r mydir
- 显示当前目录下所有压缩过的gz文件信息
\$ gzip -l *.gz
- 解压filename.gz文件
\$ gzip -d filename.gz
\$ gunzip filename.gz

zip/unzip

- 与windows下的 **winzip**兼容
- 由官方仓库的**zip/unzip**软件包提供
- 例如:
 - 压缩文件 **file1**为 **file1.zip**, 原文件保留
\$ zip file1.zip file1
 - 将子目录 **data1/** 下的所有文件压缩到文件 **data1.zip**
\$ zip -r data1.zip data1
 - 解压释放压缩文件 **data1.zip** 中的所有文件
\$ unzip data1.zip

2 文件压缩打包（tar）

- 2. tar
- tar可以对文件和目录进行打包。
- tar命令的使用语法格式如下：
 - tar [主选项+辅选项] 文件或者目录

2 文件压缩打包（tar）

下表列出了tar命令的主选项的常用选项

常用选项	说 明
-c	创建新的档案文件。如果用户想备份一个目录或是一些文件，就要选择这个选项。
-x	从档案文件中释放文件。
-t	列出档案文件的内容，查看已经备份了哪些文件。
-u	更新文件。就是说，用新增的文件取代原备份文件，如果在备份文件中找不到要更新的文件，则把它追加到备份文件的最后。

2 文件压缩打包（tar）

下表列出了tar命令的辅选项的常用选项

常用选项	说明
f	使用档案文件或设备，这个选项通常是必选的。
K	保存已经存在的文件。例如我们把某个文件还原，在还原的过程中，遇到相同的文件，不会进行覆盖。
M	创建多卷的档案文件，以便在几个磁盘中存放。
v	详细报告tar处理的文件信息。如无此选项，tar不报告文件信息
z	用gzip来压缩/解压缩文件

2 文件压缩打包 (tar)

- tar用法举例:
- `$ tar -cvf myball.tar somedirname`
- `$ tar -tf myball.tar`
- `$ tar -xvf myball.tar`

- `$ tar -zcvf myball.tar.gz somedirname`
- `$ tar -ztf myball.tar.gz`
- `$ tar -zxvf myball.tar.gz`

2 文件压缩打包（unzip）

■ 3、unzip命令

- 可以用unzip命令，该命令用于解扩展名为“.zip”的压缩文件。该命令使用的语法格式如下：

- **unzip [选项] 压缩文件名.zip**

■ unzip命令各选项的说明如表所示。

表 upzip命令的常用选项

常用选项	说 明
-x	文件列表 解压缩文件，但不包括指定的file文件
-v	查看压缩文件目录，但不解压。
-t	测试文件有无损坏，但不解压。
-d	目录把压缩文件解到指定目录下。
-z	只显示压缩文件的注解。
-n	不覆盖已经存在的文件。
-o	覆盖已存在的文件且不要求用户确认。
-j	不重建文档的目录结构，把所有文件解压到同一目录下。

RPM的依赖性问题

- **rpm -ivh girlfriend**
 - 软件包无法被安装。
 - girlfriend: 依赖于: house
 - girlfriend: 依赖于: car
- **rpm -ivh house car**
 - house,car: 依赖于: money
- **rpm -ivh money**
 - money : 依赖于:

yum

- **yum = Yellow dog Updater, Modified**
 - 主要功能是更方便的添加/删除/更新**RPM**包.
 - 它能自动解决包的倚赖性问题.
 - 它能便于管理大量系统的更新问题

yum

■ yum特点

- 可以同时配置多个资源库(**Repository**)
- 简洁的配置文件(**/etc/yum.conf**)
- 自动解决增加或删除**rpm**包时遇到的倚赖性问题
- 使用方便
- 保持与**RPM**数据库的一致性

yum安装

- CentOS自带(yum-*.noarch.rpm)
 - `#rpm -ivh yum-*.noarch.rpm`
- 在第一次启用yum之前首先需要导入系统的RPM-GPG-KEY:
 - `#rpm --import /usr/share/doc/centos-release-5/RPM-GPG-KEY-CentOS-5`

yum常用命令

- 1.列出所有可更新的软件清单
命令: **yum check-update**
- 2.安装所有更新软件
命令: **yum update**
- 3.仅安装指定的软件
命令: **yum install <package_name>**
- 4.仅更新指定的软件
命令: **yum update <package_name>**
- 5.用YUM删除软件包
命令: **yum remove <package_name>**

yum查询

- 1.使用YUM查找软件包
命令: **yum search <keyword>**
- 2.列出所有可安装的软件包
命令: **yum list**
- 3.列出所有可更新的软件包
命令: **yum list updates**
- 4.列出所有已安装的软件包
命令: **yum list installed**

yum清除缓存

- 缓存目录(/var/cache/yum)
- 1.清除缓存目录下的软件包
 - yum clean packages
- 2.清除缓存目录下的 headers
 - yum clean headers
- 3.清除缓存目录下旧的 headers
 - yum clean oldheaders
- 4.清除缓存目录下的软件包及旧的headers
 - yum clean
 - yum clean all (= yum clean packages; yum clean oldheaders)

```
root@localhost:~  
[root@localhost ~]# yum check-update  
Loaded plugins: fastestmirror, security  
Loading mirror speeds from cached hostfile  
* base: mirrors.163.com  
* extras: mirrors.163.com  
* updates: mirrors.163.com  
base | 1.1 kB 00:00  
extras | 2.1 kB 00:00  
updates | 1.9 kB 00:00  
Skipping security plugin, no data  
  
coreutils.i386 5.97-34.el5_8.1 updates  
firefox.i386 10.0.3-1.el5.centos updates  
glibc.i686 2.5-81.el5_8.1 updates  
glibc-common.i386 2.5-81.el5_8.1 updates  
gnutls.i386 1.4.1-7.el5_8.2 updates  
httpd.i386 2.2.3-63.el5.centos.1 updates  
kernel.i686 2.6.18-308.1.1.el5 updates  
ksh.i386 20100621-5.el5_8.1 updates  
libpng.i386 2:1.2.10-16.el5_8 updates  
libsmbclient.i386 3.0.33-3.38.el5_8 updates  
libvorbis.i386 1:1.1.2-3.el5_7.6 updates  
libxml2.i386 2.6.26-2.1.15.el5_8.2 updates  
libxml2-python.i386 2.6.26-2.1.15.el5_8.2 updates
```

用法举例

- 例如:

- `yum -y install httpd`
- `yum -y install php`
- `yum -y install mysql-server`
- `yum -y install php-mysql`

- 或者

- `yum -y install httpd php mysql-server php-mysql`

- 支持通配符

- `yum -y install http*`

```
root@localhost:~  
[root@localhost ~]# yum install php  
Loaded plugins: fastestmirror, security  
Loading mirror speeds from cached hostfile  
* base: mirrors.163.com  
* extras: mirrors.163.com  
* updates: mirrors.163.com  
Setting up Install Process  
Resolving Dependencies  
--> Running transaction check  
--> Package php.i386 0:5.1.6-32.el5 set to be updated  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
=====
```

Package	Arch	Version	Repository	Size
---------	------	---------	------------	------

```
=====
```

Installing:

php	i386	5.1.6-32.el5	base	2.3 M
-----	------	--------------	------	-------

```
=====
```

Transaction Summary

```
=====
```

Install	1 Package(s)
Upgrade	0 Package(s)

```
=====
```

```
root@localhost:~  
[root@localhost ~]# yum remove php  
Loaded plugins: fastestmirror, security  
Setting up Remove Process  
Resolving Dependencies  
--> Running transaction check  
--> Package php.i386 0:5.1.6-32.el5 set to be erased  
--> Finished Dependency Resolution  
  
Dependencies Resolved  
  
=====
```

Package	Arch	Version	Repository	Size
---------	------	---------	------------	------

```
=====
```

Removing:

php	i386	5.1.6-32.el5	installed	5.9 M
-----	------	--------------	-----------	-------

Transaction Summary

```
=====
```

Remove	1 Package(s)
Reinstall	0 Package(s)
Downgrade	0 Package(s)

Is this ok [y/N]: █