

安装软件包

1、安装软件包

```
# rpm -ivh /mnt/cdrom/Packages/zsh-4.3.10-5.el6.x86_64.rpm
```

-i: --install 安装

-v: --verbose 显示安装过程

-h: --human 以人类可读的方式显示

--test: 只是测试安装，并不会真正安装软件

```
[hostname Packages]# rpm -ivh --test gcc-4.4.7-3.el6.x86_64.rpm
```

--nodeps: 只安装软件本身，不考虑依赖

--force: 强制安装

2、rpm的查询： 重点****

1) -q query—查询 查询已安装的软件包的信息

a) 查询软件包是否安装 rpm -q 软件名字

```
[hostname Packages]# rpm -q zsh
```

```
zsh-4.3.10-5.el6.x86_64
```

```
[hostname Packages]# rpm -q asdf
```

```
package asdf is not installed
```

b) 查询系统中所有已经安装的软件包有哪些

```
[hostname Packages]# rpm -qa
```

统计已经安装的软件包的个数

```
[hostname Packages]# rpm -qa | wc -l
```

```
1120
```

c) f 查询某个文件是由哪个软件包安装的

rpm -qf 文件的绝对路径

找出vim命令是由哪个软件包安装的

```
[hostname ~]# rpm -qf `which vim`
```

```
vim-enhanced-7.2.411-1.8.el6.x86_64
```

练习: ifconfig命令是由哪个软件包安装的

```
[hostname ~]# rpm -qf `which ifconfig`
```

```
net-tools-1.60-110.el6_2.x86_64
```

/etc/passwd文件是由哪个软件包安装的

```
[hostname ~]# rpm -qf /etc/passwd
setup-2.8.14-20.el6.noarch
```

d) c 查询软件的配置文件

```
[hostname ~]# rpm -qc httpd
```

e) l 安装某个软件时，都安装了哪些文件

```
[hostname ~]# rpm -ql setup
```

f) i 查询安装的软件包的信息

```
[hostname ~]# rpm -qi setup
```

2) -qp 查询未安装的软件包的相关信息

-qpi 未安装的软件包的完整路径和包名 : 查询未安装的软件包的详细信息

```
[hostname ~]# rpm -qpi /mnt/cdrom/Packages/gcc-4.4.7-
```

3.el6.x86_64.rpm

-qpl: 如果安装未安装的软件，会安装哪些文件在系统中

-qpR: 查询未安装的软件包的依赖情况

3、rpm包的卸载

-e : erase

```
[hostname ~]# rpm -e zsh
```

```
[hostname ~]# rpm -q zsh
```

package zsh is not installed

rpm包的更新

-U : 如果包未安装，等同于-i 进行安装

如果包已安装，便是更新版本

4、rpm包的校验: verify

-V: 验证软件包生成的文件或命令是否被人修改过 —— 为了安全，防止恶意代码

rpm -V 已安装的软件包的名字

```
[hostname ~]# rpm -V setup
```

.....T. c /etc/bashrc

S.5....T. c /etc/printcap

S.5....T. c /etc/profile

rpm -Va 检测所有的包

S file Size differs 大小改变

M Mode differs (includes permissions and file type) 文件的类型或权限改变

5 digest (formerly MD5 sum) differs md5值改变

D Device major/minor number mismatch 设备的主从号改变

L readLink(2) path mismatch 符号链接文件找不到链接对象

U User ownership differs 用户改变

G Group ownership differs 组改变

T mTime differs 修改文件内容的时间改变

P caPabilities differ 兼容性改变

5、rpm包数字签名

```
[hostname ~]# rpm -ivh /mnt/cdrom/Packages/zsh-4.3.10-5.el6.x86_64.rpm
warning: /mnt/cdrom/Packages/zsh-4.3.10-5.el6.x86_64.rpm: Header V3
RSA/SHA256 Signature, key ID fd431d51: NOKEY
Preparing...
(100%##### [100%]
 1:zsh (
1%##### [100%]
```

导入数字签名(一次即可)

```
[hostname ~]# cd /mnt/cdrom/
[hostname cdrom]# rpm --import RPM-GPG-KEY-redhat-release
[hostname ~]# rpm -e zsh
[hostname cdrom]# rpm -ivh Packages/zsh-4.3.10-5.el6.x86_64.rpm
//不会再有警告
```

二.编译安装

```
ding.tar.gz 软件包
tar -zxf
cd

ppl cloog-ppl mpfr cpp gcc libstdc gcc-c++ 编译工具
./configure --prefix=/path/to/somefile 编译前的准备工作
make 编译工作
make install 编译安装
```

三.yum安装

常用yum命令:

install: Install a package or packages on your system 安装

update: Update a package or packages on your system 升级所有包, 改变软件设置和系统设置,系统版本内核都升级

remove: 移出某个软件, 后跟软件名称

search: Search package details for the given string 搜索某个软件或重要关键字

list: List a package or groups of packages列出yum所管理的软件名称与版本, 类似于rpm -qa

info: Display details about a package or group of packages, 类似于rpm -qai

clean: Remove cached data 下载文件被放置在/var/cache/yum中, 可以使用clean清除。

check-update: Check for available package updates 检查可用的更新包

Groupinfo: Display details about a package group 显示一个包组的信息

groupinstall Install the packages in a group on your system 安装一个包组在你的系统上

grouplist: List available package groups 列出可用的包组

groupremove: Remove the packages in a group from your system 移除包组

groupupdate: 升级整个软件群组

help: Display a helpful usage message 帮助

Deplist: List a package's dependencies 列出软件包的依赖关系, 在使用本地源软件包进行安装时, 可以查看依赖关系

Erase: Remove a package or packages from your system 移除软件包

localinstall: Install a local RPM安装本地的rpm软件包

makecache: Generate the metadata cache 生成数据缓存

reinstall: reinstall a package 重装软件包

upgrade: Update packages taking obsoletes into account 升级所有包, 不改变软件设置和系统设置, 系统版本升级, 内核不改变

yum history

帮助: yum history help

现在的 yum 带有历史记录功能,可以查看过往的事务,重做或回滚这些事务

重做是 “yum history redo N” ,

回滚是 “yum history undo N”

配置本地yum源

cd /etc/yum.repos.d找到以.repo结尾的文件

用vim编辑：

[name]

name=user_name(自定义)

添加baseurl=file:///mnt/cdrom(光盘挂载路径，即软件源的位置)

enabled=1

gpgcheck=0