Основы работы с менеджером пакетов RPM

Вступление

RPM (Redhat Package Manager) – это мощный менеджер пакетов, который может быть использован для сборки, установки, инспекции, проверки, обновления и удаления отдельных программных пакетов. Пакет состоит из набора файлов и информации о пакете, включающей название, версию и описание пакета.

RPM спроектирован так, что предоставляет мощную систему запросов. Вы можете искать в вашей базе данных информацию о пакете или просто об определенных файлах. Вы также можете легко определить, к какому пакету принадлежит файл, и откуда он появился. Сами файлы RPM являются сжатыми архивами, но вы можете запрашивать отдельный пакет очень легко и быстро, потому что к пакету добавлен дополнительный двоичный заголовок со всем, что вам может быть необходимо знать в несжатой форме. Это позволяет производить быстрые запросы.

Другое мощное свойство – это проверка пакетов. Если вы беспокоитесь о том, что вы удалили важный файл из некоторого пакета, просто проверьте это. Вы будете оповещены о любых аномалиях. С этой точки вы можете переустановить пакет, если это необходимо. Любые конфигурационные файлы будут сохранены.

RPM является очень мощным, гибким и удобным менеджером пакетов, который многие начинающие админы почему то игнорируют. Конечно, у него тоже есть свои недостатки, но у кого их нет? В этой статье, я попытаюсь доказать, что использование rpm облегчает, и без того не легкую, жизнь сисадмина :). А также избавляет его от многих проблем, которые возникнут при установке ПО из исходных кодов.

Из названия менеджера (Redhat Package Manager) понятно, что я буду рассматривать redhat подобные дистрибутивы CentOS, WhiteBox, Fedora Core, Mandrake, ALT Linux, ASP Linux, RHEL, SuSe и т.д., т.к. например в debian или gentoo используется совершенно другой формат пакетов и менеджеры у них соответственно другие. Хотя и на любых других дистрибутивах следует использовать их родные менеджеры пакетов.

Усвойте три самых главных правила

1. Устанавливать/удалять/обновлять все ПО, следует только при помощи rpm;
2. Если нужной вам программы нет ввиде rpm пакета, то можно собрать самому свой rpm пакет (правда, для этого необходим некоторый опыт и по началу это может быть трудным), но зато потом вы сэкономите себе очень много времени и нервов;
3. Если вы все-таки решили установить из исходников, то смотрите первых два правила

Основные операции

Пожалуй наиболее частой операцией является установка/удаление пакетов. Рассмотрим каждый из случаев.

Установка пакета с диска

```
# mount /media/cdrom
# cd /media/cdrom/CentOS/RPMS
# rpm -ivh squid-2.5.STABLE6-3.4E.3.i386.rpm
предупреждение: squid-2.5.STABLE6-3.4E.3.i386.rpm:
подпись DSA V3: NOKEY, key ID 443e1821
Подготовка...
########################################### [100%]
                                                1:squid
########################################### [100%]
```

Обратите внимание, что при установке необходимо указывать полный путь и имя rpm пакета, в отличие от удаления.
Теперь мы можем проверить установлен ли у нас пакет

```
# rpm -q squid
squid-2.5.STABLE6-3.4E.3
```

Удаление пакета

Процедура удаления выполняется также легко, как и процедура установки. Разница состоит лишь в том, что нам не надо указывать путь и полное имя пакета. Достаточно указать лишь его имя, т.е. squid, хотя если вы укажите squid-2.5.STABLE6 или squid-2.5.STABLE6-3.4E.3, то ошибки не будет и пакет также удалится.

```
# rpm -e squid
```

Вообще то при удалении, необходимо указывать имя пакета, которое хранится в базе данных rpm. Если вы не знаете точного имени, то можно воспользоваться следующей командой и выяснить имя соответствующего пакета.

```
# rpm -qa | grep -i mysql
MySQL-server-4.1.9-0
MySQL-devel-4.1.9-0
MySQL-shared-compat-4.1.9-0
MySQL-client-4.1.9-0
```

Данная команда выведет список всех пакетов, в названии которых встречается слово mysql, без учета регистра. Если вы захотите удалить, например mysql-devel, то необходимо будет полностью указывать имя пакета, т.е. MySQL-devel, а не MySQL.

Обратите внимание, что регистр здесь имеет значение. Иначе при попытке удалить пакет, вы получите следующее сообщение

```
# rpm -e mysql-devel
ошибка: пакет mysql-devel не установлен
```

А с учетом регистра пакет удалится без проблем

```
# rpm -e MySQL-devel
```

Обновление пакета. Рассмотрим случай, когда у вас в системе уже установлена предыдущая версия пакета, и вы хотите обновить

его.

```
# rpm -q squid
squid-2.5.STABLE6-3.4E.3
```

Можно конечно удалить старый пакет и установить новый. Но здесь тоже может быть не все так гладко, от данного пакета могут зависеть другие, установленные пакеты и тогда вы не сможете его удалить. Точнее сможете, но только с ключом –nodeps, который следует использовать ОЧЕНЬ ОСТОРОЖНО и только, если вы точно знаете, что делаете. Другой способ при установке воспользоваться ключом -U (upgrade).

Также полезно использовать ключ –test, с этим ключом будет имитироваться процесс установки/обновления пакета, т.е. файлы устанавливаться не будут. Если будут найдены, какие то ошибки или неудовлетворенные зависимости, то вам об этом будет выдано сообщение. Если никаких сообщений не будет, то можно спокойно устанавливать пакет

На время обновления пакета, наверное, лучше остановить squid во избежание конфликтов

```
# service squid stop
Останавливается squid: .                                   [
OK  ]
```

Производим непосредственное обновление пакета.

```
# rpm -Uvh squid-2.5.STABLE11-2.athlon.rpm
Подготовка...
########################################### [100%]
   1:squid
     предупреждение: /etc/squid/squid.conf создан как
/etc/squid/squid.conf.rpmnew
########################################### [100%]
```

Из сообщения мы видим, чтобы не перезаписать наш текущий конфигурационный файл – **squid.conf**, rpm сохранил новый файл под именем **squid.conf.rpmnew**. Все же перед обновлением рекомендую копировать текущие конфигурационные файлы в

отдельную папку во избежание всякого рода неприятностей.

Проверяем версию squid

```
# rpm -q squid
squid-2.5.STABLE11-2
```

Как видим, squid у нас успешно обновился. Запускаем squid

```
# service squid start
Запускается squid: .                                        [
OK  ]
```

Узнать список всех установленных пакетов в системе можно с помощью следующей команды

```
# rpm -qa
redhat-logos-1.1.25-1.centos4.3
basesystem-8.0-4
glibc-2.3.4-2
chkconfig-1.3.11.2-1

...
...
...

apache-1.3.33-1
cyrus-sasl-plain-2.1.20-4
openssl-0.9.7e-3
curl-devel-7.13.1-1
```

Также еще одной полезной возможностью является проверка целостности пакета. Вы в любой момент времени можете проверить, все ли файлы существуют в системе.

```
# rpm -V squid
```

Если не будет никаких сообщений, то значит, целостность пакета не нарушена. Для проверки работоспособности данной функции я специально удалю два файла принадлежащих пакету squid и выполню проверку пакета еще раз.

```
# rm -f /etc/squid/squid.conf /etc/squid/mime.conf
```

```
# rpm -V squid
missing   c /etc/squid/mime.conf
missing   c /etc/squid/squid.conf
```

Как видно из сообщения он указывает, что отсутствуют два файла, которые я удалил. Если вы хотите проверить целостность всех пакетов, установленных в системе, то необходимо указать ключ -Va.

Еще одной полезной функцией, предоставляемой rpm, является отображение информации о пакете. Допустим, вы хотите узнать, для чего предназначен этот пакет не устанавливая его. Получить информацию о пакете можно с помощью следующей команды

```
# rpm -qpi MySQL-devel-4.1.9-0.i386.rpm
Name        : MySQL-devel                  Relocations: (not relocatable)
Version     : 4.1.9                        Vendor: MySQL AB
Release     : 0                            Build Date: Чтв 13 Янв 2005 06:35:22
Install Date: (not installed)              Build Host: build.mysql.com
Group       : Applications/Databases       Source RPM: MySQL-4.1.9-0.src.rpm
Size        : 6912539                      License: GPL
Signature   : DSA/SHA1, Чтв 13 Янв 2005 17:06:48, Key ID 8c718d3b5072e1f5
Packager    : Lenz Grimmer
URL         : http://www.mysql.com/
Summary     : MySQL - Development header files and libraries
Description :
This package contains the development header files and libraries
necessary to develop MySQL client applications.

For a description of MySQL see the base MySQL RPM or http://www.mysql.com
```

Также с помощью rpm вы можете получить список файлов принадлежащих данному пакету, а также пути, куда они были установлены. Для получения данной информации необходимо

выполнить следующую команду

```
# rpm -ql rpm-build
/usr/bin/rpmbuild
/usr/lib/rpm/brp-compress
/usr/lib/rpm/brp-python-bytecompile
/usr/lib/rpm/brp-redhat
/usr/lib/rpm/brp-sparc64-linux

...
...
...

/usr/src/redhat/RPMS/i686
/usr/src/redhat/RPMS/noarch
/usr/src/redhat/SOURCES
/usr/src/redhat/SPECS
/usr/src/redhat/SRPMS
```

Если вы хотите просмотреть список файлов, предоставляемых данным пакетом, но при этом не хотите устанавливать сам пакет, то выполните следующую команду.

```
# rpm -qpl /media/cdrom/CentOS/RPMS/linuxdoc-tools-0.9.20-14.i386.rpm
/usr/bin/linuxdoc
/usr/bin/rtf2rtf
/usr/bin/sgml2html
/usr/bin/sgml2info
/usr/bin/sgml2latex

...
...
...

/usr/share/man/man1/sgmlsasp.1.gz
/usr/share/texmf/tex/latex/misc/linuxdoc-sgml.sty
/usr/share/texmf/tex/latex/misc/linuxdoctr-sgml.sty
/usr/share/texmf/tex/latex/misc/null.sty
/usr/share/texmf/tex/latex/misc/qwertz.sty
```

Иногда, бывает необходимо узнать, к какому пакету принадлежит

определенный файл и здесь вам опять поможет rpm. Для получения этой информации необходимо выполнить следующую команду.

```
# rpm -qf /usr/sbin/dmidecode
kernel-utils-2.4-13.1.48
```

Из вывода видно, что файл dmidecode принадлежит пакету kernel-utils-2.4-13.1.48. Хоть я и перешел на freebsd, но все-таки система портов, к сожалению пока не предоставляет таких гибких средств управления пакетами и запросами, как rpm.

А теперь, представьте себе, что вы устанавливали бы все из исходников, при этом вы автоматически лишаетесь всех вышеописанных преимуществ. Для того, чтобы удалить необходимую программу вам необходимо помнить куда вы ее поставили и потом вручную удалять все файлы. При установке программы вам каждый раз придется собирать ее заново.

При конфигурировании программы можно воспользоваться ключом –prefix и указать путь для установки, но это тоже не является выходом из ситуации, т.к. все файлы будут установлены в одну директорию. Конечно, можно воспользоваться и другими ключами, такими как –mandir, –datadir, –infodir, –libdir, –includedir и другими. Но я считаю, что это не лучший выход из ситуации.

При использовании rpm всех этих проблем не существует и все «части» программы (конфигурационные файлы, исполняемые файлы, файлы помощи, скрипты и т.д.) будут установлены в соответствующие папки (прочитайте man hier для ознакомления с иерархией файловой системы). А обновление программ, установленных из исходников, превращается в настоящую муку.

Также, очень часто, при установке программ из исходников вы лишаетесь еще одного преимущества – это скрипты, для автоматического запуска программ при загрузке ОС, которые располагаются в папке /etc/init.d/. Управлять автозагрузкой необходимых программ очень удобно с помощью утилиты chkconfig. В принципе можно и самому написать необходимый скрипт, но зачем это делать, когда это уже сделали для вас?

Немного безопасности не помешает

В целях безопасности перед установкой или обновлением пакетов следует проверять их на подлинность, чтобы быть уверенным, что вы действительно устанавливаете необходимый вам пакет. Для этого существует два способа – проверка GPG подписи и md5 суммы. Рассмотрим каждый из них.

Для того, чтобы проверить GPG подпись необходимо вызвать rpm с ключом –check-sig.

```
# mount /media/cdrom
# cd /media/cdrom/CentOS/RPMS/
# rpm --checksig squid-2.5.STABLE6-3.4E.3.i386.rpm
squid-2.5.STABLE6-3.4E.3.i386.rpm: (SHA1) DSA sha1 md5 (GPG) НЕ OK
(ОТСУТСТВУЮТ КЛЮЧИ: GPG#443e1821)
```

Как видно из сообщения, у нас отсутствует открытый ключ GPG, и соответственно мы не можем проверить подпись. Для исправления этой ситуации необходимо импортировать ключ GPG в нашу базу данных с помощью следующей команды

```
# rpm --import /media/cdrom/RPM-GPG-KEY
```

На всех дисках с дистрибутивами должен идти GPG ключ, с помощью которого мы можем проверить все пакеты входящие в этот дистрибутив. После импорта ключа повторим проверку rpm пакета.

```
# rpm --checksig squid-2.5.STABLE6-3.4E.3.i386.rpm
squid-2.5.STABLE6-3.4E.3.i386.rpm: (sha1) dsa sha1 md5 gpg OK
```

Как видно, из сообщения, у нас действительно оригинальный (не измененный) пакет и его можно безопасно устанавливать. Но что делать, если вы скачиваете пакет с сайта производителя ПО? Очень часто производители также выкладывают открытые GPG ключи на сайтах. В данном случае необходимо выполнить аналогичную процедуру, т.е. импортировать ключ в базу данных, а затем проверить соответствующий пакет.

Если все-таки нет GPG ключа, то проверить сохранность пакета

можно по т.н. контрольной сумме. Перед установкой пакета, следует сравнить текущую сумму с указанной на сайте, откуда вы скачивали данный rpm пакет. Узнать контрольную сумму можно с помощью следующей команды

```
# md5sum postfix-2.2.8-1.2.src.rpm
985be6c349f062c5245a5a3272c2ac2c  postfix-2.2.8-1.2.src.rpm
```

Если md5 сумма совпадает, значит у вас действительно оригинальный пакет и теперь можно приступить к его непосредственной установке. Если сумма не совпадает, то возможно при закачке файл был поврежден или это совсем не тот файл, который вам нужен ;). В данном случае лучше не устанавливать файл. Если у вас вообще нет никакой возможности проверить подлинность того или иного пакета, то я рекомендовал бы не устанавливать данный пакет вообще. Иначе при установке такого пакета вы можете установить себе в систему т.н. rootkit о последствиях я думаю говорить не надо :).

Надеюсь мне удалось убедить вас, что rpm является действительно мощным, гибким и удобным менеджером пакетов. Используя который ваша система всегда будет согласована и работать, как часы.

А когда после прочтения второй части статьи вы научитесь собирать свои собственные пакеты, то думаю, что вы полюбите его еще больше, и будете использовать только rpm (или ваш штатный менеджер пакетов).

Репозитарии

- RPMforge
- epel
- Remi
- atrpms
- ELRepo
- IUS Community
- Webtatic

Сайты для поиска rpm

- rpmfind
- dag wieers
- rpmseek