

Основы системного администрирования и сетевых технологий

УРОК №9

Способы установки программ

В Linux существует несколько способов установки программ.

- 1) Программа распространяется в виде архива исходных кодах *.tar.gz, которые необходимо распаковать и, в простейшем случае, откомпилировать с помощью команд `make`, `make install`.
- 2) Воспользоваться пакетным менеджером и пакетами с исходным кодом.
- 3) Воспользоваться менеджером пакетов и установить программу, которая распространяется в виде пакета откомпилированных файлов.

Менеджеры пакетов

Менеджеры пакетов – это инструменты, используемые для инсталляции и сопровождения программного обеспечения. Они помогают отслеживать, какие установлены программы и где располагаются файлы.

Для отслеживания установленных приложений менеджер пакетов создает централизованную базу данных. Наибольшее распространение получили два формата пакетов: RPM и Debian.

- RPM** RPM (RPM Package Manager) предоставляет следующие возможности:
- установка новых программ;
 - удаление программ;
 - проверка – позволяет убедиться в том, что программные файлы не повреждены и не искажены;
 - модернизация версий установленных программ;
 - запрос установленного программного обеспечения, например, ответ на вопрос «Из какого пакета установлен этот файл?»;
 - извлечение – проверка содержимого пакетов перед установкой.

Принципы именования пакетов rpm

Имя пакета характеризует сам пакет, его версию, версию сборки исполняемых файлов (релиз) и архитектуру и задается в виде «имя_программы-версия-релиз.платформа» или «src.rpm».

В табл. 1 представлены наиболее распространенные названия архитектур, используемые в пакетах rpm.

Таблица 1.
Распространенные названия
архитектур в пакетах rpm

Метка	Описание
i386	Служит указателем на совместимость с процессором 80386. Для работы требуется процессор минимум Pentium I
i486	Пакет поддерживается процессорами 80486
i586	Пакет поддерживается всеми процессорами Pentium и более новыми
i686	Для работы требуется процессор Pentium II и выше
ix86	Для работы требуется процессор Pentium и выше
x86_64	Процессоры AMD Орегон и Pentium 4 с поддержкой инструкций EM64T. Пакет с такой меткой скомпилирован для работы в 64 битном режиме, то есть не совместим с 32 битными процессорами. Также не будет работать под управлением 32 битного ядра Linux
IA64	Метка указывает на 64 разрядные процессоры Itanium

Информация, содержащаяся в пакете

Полный формат грт-файла - это четыре секции.

1) Начальный идентификатор, помечающий файл как грт-пакет. Секция идентификатора создается соответствующей версией грт, которая использовалась для сборки.

2) Сигнатура следует за начальным идентификатором. Подобно любой цифровой подписи сигнатура позволяет оценить впоследствии целостность пакета. Сигнатура является результатом выполнения некоторой функции над заголовком и нагрузкой. Это может быть криптофункция, например PGP или хэш-функция, например дайджест MD5.

3) Заголовок содержит ноль или более полей данных, описывающих пакет. В служебные поля заключается информация о правах на программное обеспечение, версия, описание пакета и некоторые другие сведения.

4) Нагрузка содержит файлы, которые используются в проекте. Когда отдана команда на установку пакета, эти файлы копируются в систему в нужные места. Все дерево каталогов и файлов пакета сжато с помощью gzip.

Бинарные гртм и гртм с исходным кодом

Существует два типа пакетов гртм - бинарные и с исходным кодом.

Бинарные пакеты

Бинарные пакеты содержат полный набор приложений и библиотек, собранных для определенной процессорной архитектуры. Работа этих приложений часто зависит от других приложений (библиотек), которые в свою очередь содержатся в иных бинарных гртм-пакетах.

Хотя большая часть пакетов включает приложения, некоторые пакеты содержат только библиотеки функций. Пакеты библиотек позволяют поддерживать работу множества приложений с помощью одной библиотеки (shared libraries). Поэтому, как правило, библиотеки входят в пакеты, не содержащие приложений.

В дополнение к бинарным пакетам для определенных архитектур, гртм поддерживает концепцию независимых от платформы бинарных пакетов (poarch-пакетов), предоставляющих библиотеки или коллекции сценариев (скриптов), работа которых не зависит от конкретной платформы. Приложения, написанные на Perl, Python или shell не связаны необходимостью компиляции. Многие приложения Java также не зависят от платформы.

Пакеты с исходным кодом

В соответствии с соглашениями об именовании, имя файла пакета с исходным кодом оканчивается постфиксом `src.gpm`. `src.gpm`-пакет содержит средства, необходимые для сборки бинарного пакета. Наличие такого пакета означает, что можно пересобрать бинарный пакет, сделав необходимые изменения или исправления, либо для другой процессорной архитектуры.

Заголовок пакета

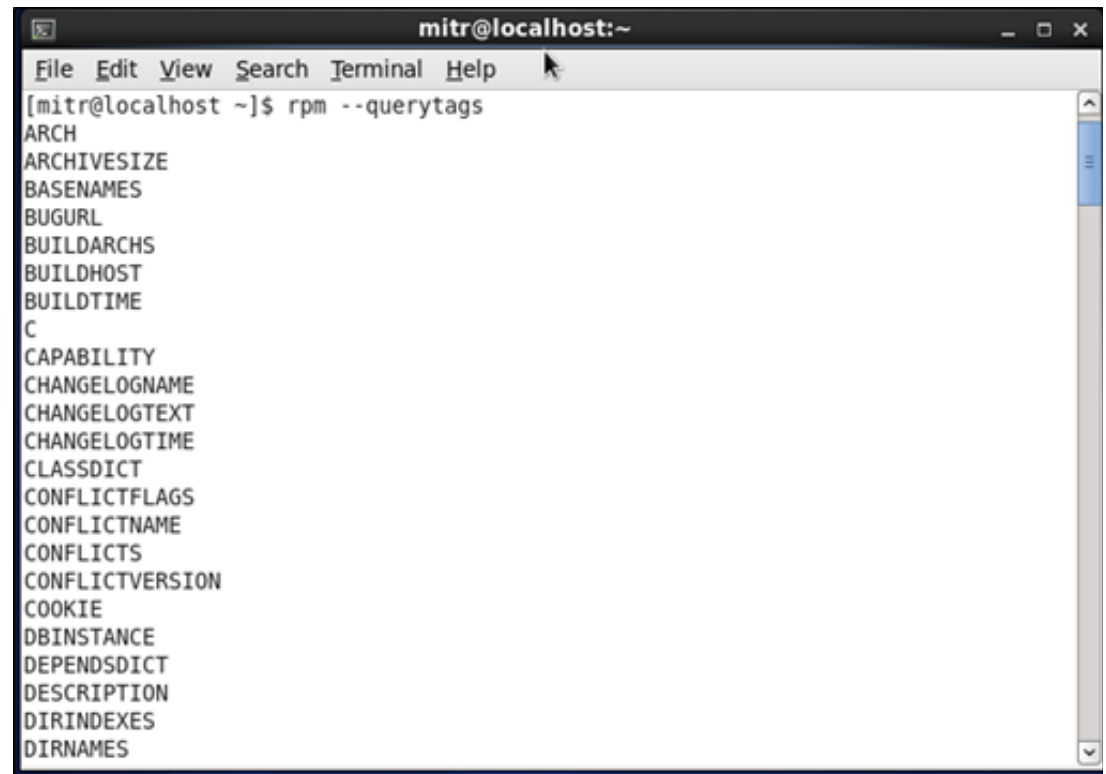
Заголовок содержит теги, описывающие пакет. В них содержится информация о правах на программное обеспечение, версия, описание пакета и некоторые другие сведения. Для пользователя обычно наиболее интересны следующие теги:

- `name` – имя программы;
- `version` – версия программы;
- `release` – версия сборки;
- `license` – лицензия, по которой распространяется пакет;
- `summary` – краткое описание пакета;
- `description` – описание пакета;
- `group` – группа/подгруппа программного обеспечения;
- `vendor` – сборщик пакета.

Получить список тегов можно с помощью команды, показанной на рис. 1.

Содержимое `group` указывает, к какой категории программного обеспечения относится пакет.

Рисунок 1.
Получение списка тегов заголовка пакета.



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm --querytags  
ARCH  
ARCHIVESIZE  
BASENAMES  
BUGURL  
BUILDARCHS  
BUILDHOST  
BUILDTIME  
C  
CAPABILITY  
CHANGELOGNAME  
CHANGELOGTEXT  
CHANGELOGTIME  
CLASSDICT  
CONFLICTFLAGS  
CONFLICTNAME  
CONFLICTS  
CONFLICTVERSION  
COOKIE  
DBINSTANCE  
DEPENDSDICT  
DESCRIPTION  
DIRINDEXES  
DIRNAMES
```


Нагрузка пакета

Нагрузка содержит файлы, которые реально используются в проекте. Когда отдана команда на установку пакета, эти файлы копируются в систему в нужные места. Все дерево каталогов и файлов пакета архивировано и сжато с помощью gzip.

База данных rpm

База данных RPM хранит информацию о всех установленных пакетах. Эта база может использоваться для запросов, касающихся того, что установлено, для информирования о версиях установленного ПО, для оценки целостности пакетов и системы. Файлы базы данных располагаются в каталоге `/var/lib/rpm` (рис. 2).

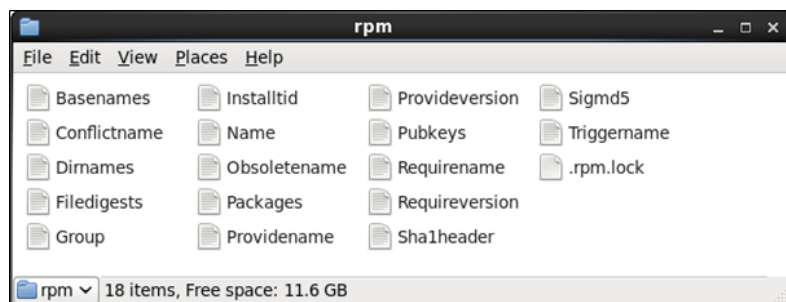


Рисунок 2. Файлы базы данных rpm

Наиболее значимый файл - `Packages`. В нем хранятся значения полей заголовка для каждого пакета, индексированные по внутреннему номеру пакета. Другие файлы предназначены для оптимизации запросов и ускорения работы базы данных.

Режим работы	Короткая нотация	Длинная нотация
Обновление/установка	-U	--upgrade
Установка	-i	--install
Удаление	-e	--erase
Режим запросов	-q	--query
Верификация	-V	--verify
Проверка подписи	-K	--checking
Обновление в режиме freshen	-F	--freshen
Инициализация БД	Нет	--initdb
Перестройка БД	Нет	--rebuilddb

Таблица 2.
Опции грм для основных режимов работы

Использование консольного менеджера грм
Консольный менеджер пакетов грм может использоваться для:

- инициализации и пересборки базы данных;
- запроса к базе данных установленных пакетов;
- установки, обновления, удаления пакетов;
- верификации и проверки подписи пакета;
- сборки пакетов.

В таблице 2 приводится список основных режимов работы грм.

Опции запроса к базе данных установленных пакетов

Общая форма команды запроса к базе данных установленных пакетов:

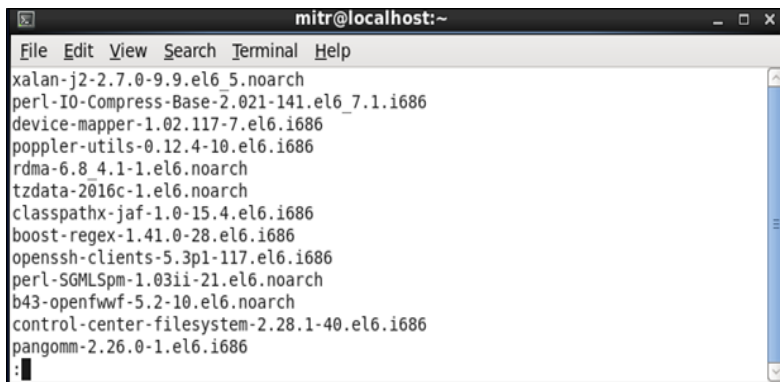
`grm -q [опции_запроса]`

Существуют два набора опций для запроса – выбор пакетов и выбор информации.

Для запроса установленного пакета, имеющего имя <имя-пакета> используется команда `grm -q` со следующими опциями:

- `a` (или `-all`) – запрос всех установленных пакетов;
 - `whatrequires <capability>` – запрос всех пакетов, требующих `<capability>` для правильного функционирования;
 - `whatrequires <virtual>` – запрос всех пакетов, предоставляющих сервис `<virtual>`;
 - `f <файл>` (или `-file <файл>`) – запрос пакета, которому принадлежит файл `<файл>`;
 - `g <группа>` (или `-group <группа>`) – запрос пакетов из группы `<группа>`;
 - `p <файл_пакета>` – запрос еще не установленного пакета `<файл_пакета>`. Файл пакета может быть задан как FTP- или HTTP-адрес;
 - `querybynumber <num>` – запросить непосредственно запись базы данных с номером `<num>`. Используется для отладочных целей;
 - `triggeredby <имя_пакета>` – запрос всех пакетов, содержащих триггер-скрипты, активизируемые пакетом `<имя_пакета>`.
- Для выбора информации используются следующие опции:
- `i` – выводит информацию о пакете, включая название, версию и описание. Использует формат вывода `-queryformat`, если он задан;

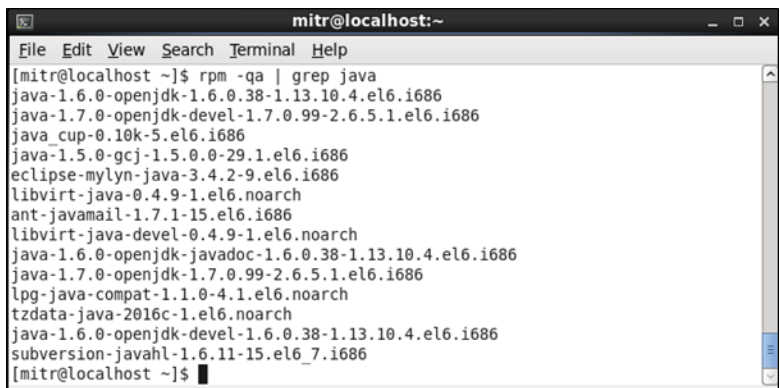
- provides – выводит список сервисов и библиотек, предоставляемых указанным пакетом;
- changelog – выводит протокол изменений указанного пакета;
- l (или -list) – выводит список файлов, входящих в данный пакет;
- s (или -state) – выводит список файлов в пакете (подразумевает -l). Каждый файл может находиться в одном из следующих состояний: нормальный, не установлен или заменен;
- d (или -docfiles) – выводит только список файлов документации (подразумевается -l);
- c (или -configfiles) – выводит список только конфигурационных файлов (подразумевает -l);
- scripts – выводит для заданного пакета сценарии (скрипты), используемые как часть процессов инсталляции или деинсталляции, если таковые есть;
- triggers (или -triggerscripts) – показать все триггер-скрипты, содержащиеся в пакете, если они имеются;
- dump – выводит информацию о файлах следующим образом: path size mtime md5sum mode owner group isconfig r dev symlink. Эта опция должна использоваться в сочетании, по меньшей мере, с одной из опций –l, -c или –d;
- last – упорядочивает список пакетов по времени установки таким образом, что наиболее свежие пакеты находятся в начале списка;
- filesbypkg – показывает все файлы в каждом пакете;
- triggerscripts – показывает триггер-скрипты для выбранных пакетов.



```
mitr@localhost:~
File Edit View Search Terminal Help
xalan-j2-2.7.0-9.9.el6_5.noarch
perl-IO-Compress-Base-2.021-141.el6_7.1.i686
device-mapper-1.02.117-7.el6.i686
poppler-utils-0.12.4-10.el6.i686
rdma-6.8.4-1-1.el6.noarch
tzdata-2016c-1.el6.noarch
classpathx-jaf-1.0-15.4.el6.i686
boost-regex-1.41.0-28.el6.i686
openssh-clients-5.3p1-117.el6.i686
perl-SGMLSpm-1.03ii-21.el6.noarch
b43-openfwf-5.2-10.el6.noarch
control-center-filesystem-2.28.1-40.el6.i686
pangomm-2.26.0-1.el6.i686
:
```

Рисунок 3.

Просмотр установленных пакетов



```
mitr@localhost:~
File Edit View Search Terminal Help
[mitr@localhost ~]$ rpm -qa | grep java
java-1.6.0-openjdk-1.6.0.38-1.13.10.4.el6.i686
java-1.7.0-openjdk-devel-1.7.0.99-2.6.5.1.el6.i686
java_cup-0.10k-5.el6.i686
java-1.5.0-gcj-1.5.0-29.1.el6.i686
eclipse-mylyn-java-3.4.2-9.el6.i686
libvirt-java-0.4.9-1.el6.noarch
ant-javamail-1.7.1-15.el6.i686
libvirt-java-devel-0.4.9-1.el6.noarch
java-1.6.0-openjdk-javadoc-1.6.0.38-1.13.10.4.el6.i686
java-1.7.0-openjdk-1.7.0.99-2.6.5.1.el6.i686
lpg-java-compat-1.1.0-4.1.el6.noarch
tzdata-java-2016c-1.el6.noarch
java-1.6.0-openjdk-devel-1.6.0.38-1.13.10.4.el6.i686
subversion-javahl-1.6.11-15.el6_7.i686
[mitr@localhost ~]$
```

Рисунок 4.

Получение имен пакетов,
в которых присутствует «java»

Примеры использования опций запросов к базе данных пакетов

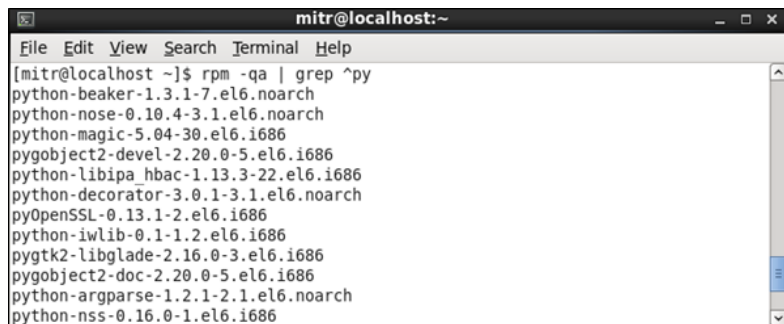
Чтобы запросить информацию обо всех установленных пакетах можно использовать команду

`rpm -qa`

Список пакетов, как правило, получается длинным, поэтому лучше использовать вызов, который передаст результаты запросы команде `less`, которая позволяет прокручивать результаты вывода (рис. 3)

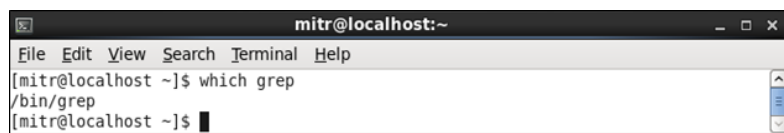
`rpm -qa | less`

Если требуется отобразить только определенные пакеты, то удобно передавать вывод команды запроса к базе данных пакета команде `grep`. Например, команда, показанная на рис. 4, выводит имена всех установленных пакетов, в которых присутствует «java».



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qa | grep ^py  
python-beaker-1.3.1-7.el6.noarch  
python-nose-0.10.4-3.1.el6.noarch  
python-magic-5.04-30.el6.i686  
pygobject2-devel-2.20.0-5.el6.i686  
python-libipa_hbac-1.13.3-22.el6.i686  
python-decorator-3.0.1-3.1.el6.noarch  
pyOpenSSL-0.13.1-2.el6.i686  
python-iwlib-0.1-1.2.el6.i686  
pygtk2-libglade-2.16.0-3.el6.i686  
pygobject2-doc-2.20.0-5.el6.i686  
python-argparse-1.2.1-2.1.el6.noarch  
python-nss-0.16.0-1.el6.i686
```

Рисунок 5.
Получение имен пакетов, имена
которых начинаются на «ру»



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ which grep  
/bin/grep  
[mitr@localhost ~]$
```

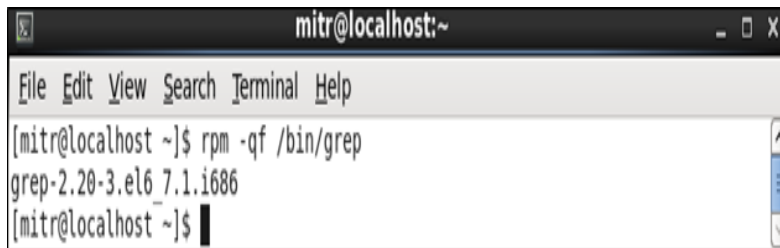
Рисунок 6.
Определение полного имени файла

Для поиска пакета, имя которого начинается с определенного сочетания символов, можно использовать следующую команду, где `ру` - начальные символы имен пакетов.

Одной из наиболее общих задач управления пакетами является задача отслеживания принадлежности определенных файлов пакетам. Для решения этой задачи используются опции `-qf`:
`rpm -qf полное_имя_файла`

Сначала необходимо определить полный путь файла. Для этого часто используется команда `which`. На рис. 6 показан способ определения полного имени файла с именем `grep`.

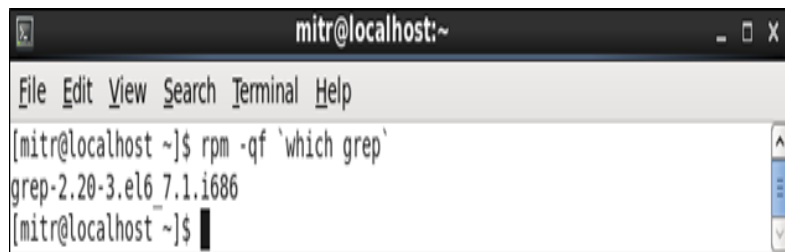
Теперь можно определить принадлежность `grep` пакету (рис. 7).



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qf /bin/grep  
grep-2.20-3.el6_7.1.i686  
[mitr@localhost ~]$
```

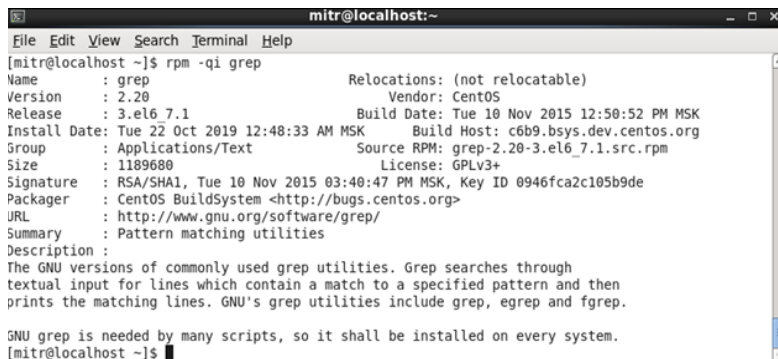
Рисунок 7.
Определение принадлежности
файла пакету

Можно объединить последовательность команд в одну команду, как это показано на рис. 8.



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qf `which grep`  
grep-2.20-3.el6_7.1.i686  
[mitr@localhost ~]$
```

Рисунок 8.
Определение принадлежности файла
пакету в одной строке

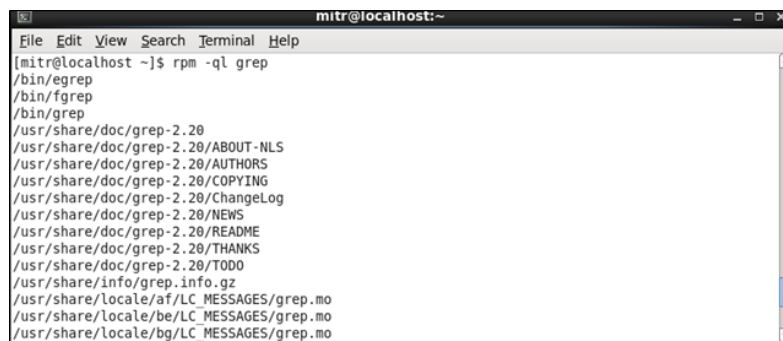


```
mitr@localhost:~$ rpm -qi grep
Name      : grep                      Relocations: (not relocatable)
Version   : 2.20                      Vendor: CentOS
Release   : 3.el6_7.1               Build Date: Tue 10 Nov 2015 12:50:52 PM MSK
Install Date: Tue 22 Oct 2019 12:48:33 AM MSK   Build Host: c6b9.bsys.dev.centos.org
Group     : Applications/Text        Source RPM: grep-2.20-3.el6_7.1.src.rpm
Size      : 1189680                  License: GPLv3+
Signature : RSA/SHA1, Tue 10 Nov 2015 03:40:47 PM MSK, Key ID 0946fca2c105b9de
Packager  : CentOS BuildSystem <http://bugs.centos.org>
URL       : http://www.gnu.org/software/grep/
Summary   : Pattern matching utilities
Description:
The GNU versions of commonly used grep utilities. Grep searches through
textual input for lines which contain a match to a specified pattern and then
prints the matching lines. GNU's grep utilities include grep, egrep and fgrep.

GNU grep is needed by many scripts, so it shall be installed on every system.
[mitr@localhost ~]$
```

Рисунок 9.

Получение информации о пакете грег



```
mitr@localhost:~$ rpm -ql grep
/bin/egrep
/bin/fgrep
/bin/grep
/usr/share/doc/grep-2.20
/usr/share/doc/grep-2.20/ABOUT-NLS
/usr/share/doc/grep-2.20/AUTHORS
/usr/share/doc/grep-2.20/COPYING
/usr/share/doc/grep-2.20/ChangeLog
/usr/share/doc/grep-2.20/NEWS
/usr/share/doc/grep-2.20/README
/usr/share/doc/grep-2.20/THANKS
/usr/share/doc/grep-2.20/TODD
/usr/share/info/grep.info.gz
/usr/share/locale/af/LC_MESSAGES/grep.mo
/usr/share/locale/be/LC_MESSAGES/grep.mo
/usr/share/locale/bg/LC_MESSAGES/grep.mo
```

Рисунок 10.

Файлы пакета грег (сокращено)

Если файл не принадлежит пакету, будет выведено сообщение «file ... is not owned by any package».

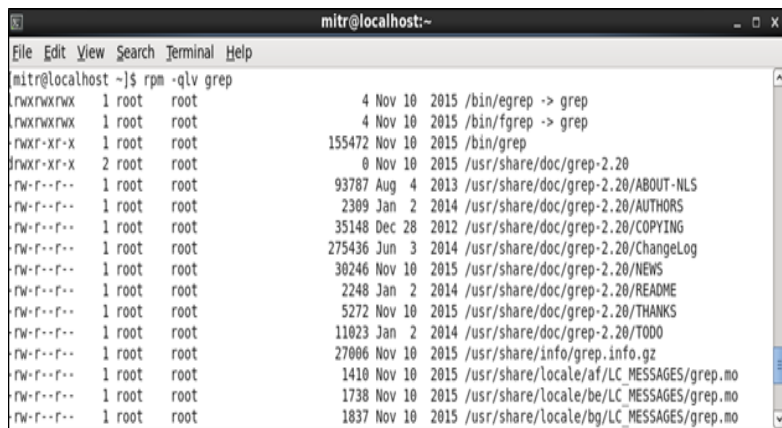
Для получения информации о пакете используются опции -qi. Последовательность указания опций важна. Если -i будет на первом месте, утилита перейдет в режим установки пакета. В запросы в качестве имени пакета допустимо включать:

- короткое имя пакета,
- имя пакета с версией,
- имя пакета с версией и номером сборки.

Также в командной строке запроса может присутствовать не одно имя пакета, а несколько, введенные через пробелы.

Это описание достаточно информативно. Оно включает даже полное имя пакета с исходным кодом, из которого был создан данный пакет. В нашем случае это пакет грег-2.20-3.el6_7.1.src.rpm.

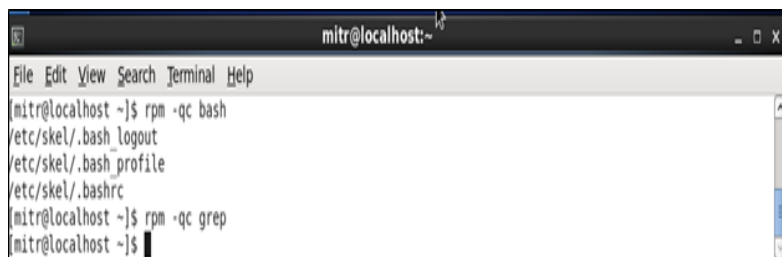
С помощью опций -ql можно получить список всех файлов пакета (рис. 10).



```
mitr@localhost:~$ rpm -qlv grep
lrwxrwxrwx 1 root root      4 Nov 10 2015 /bin/egrep -> grep
lrwxrwxrwx 1 root root      4 Nov 10 2015 /bin/fgrep -> grep
-rwxr-xr-x 1 root root 155472 Nov 10 2015 /bin/grep
-rwxr-xr-x 2 root root      0 Nov 10 2015 /usr/share/doc/grep-2.20
-rw-r--r-- 1 root root 93787 Aug 4 2013 /usr/share/doc/grep-2.20/ABOUT-NLS
-rw-r--r-- 1 root root 2389 Jan 2 2014 /usr/share/doc/grep-2.20/AUTHORS
-rw-r--r-- 1 root root 35148 Dec 28 2012 /usr/share/doc/grep-2.20/COPYING
-rw-r--r-- 1 root root 275436 Jun 3 2014 /usr/share/doc/grep-2.20/ChangeLog
-rw-r--r-- 1 root root 38246 Nov 10 2015 /usr/share/doc/grep-2.20/NEWS
-rw-r--r-- 1 root root 2248 Jan 2 2014 /usr/share/doc/grep-2.20/README
-rw-r--r-- 1 root root 5272 Nov 10 2015 /usr/share/doc/grep-2.20/THANKS
-rw-r--r-- 1 root root 11823 Jan 2 2014 /usr/share/doc/grep-2.20/TODO
-rw-r--r-- 1 root root 27006 Nov 10 2015 /usr/share/info/grep.info.gz
-rw-r--r-- 1 root root 1410 Nov 10 2015 /usr/share/locale/af/LC_MESSAGES/grep.mo
-rw-r--r-- 1 root root 1738 Nov 10 2015 /usr/share/locale/be/LC_MESSAGES/grep.mo
-rw-r--r-- 1 root root 1837 Nov 10 2015 /usr/share/locale/bg/LC_MESSAGES/grep.mo
```

Рисунок 11.
Вывод расширенной информации
о файлах пакета грег

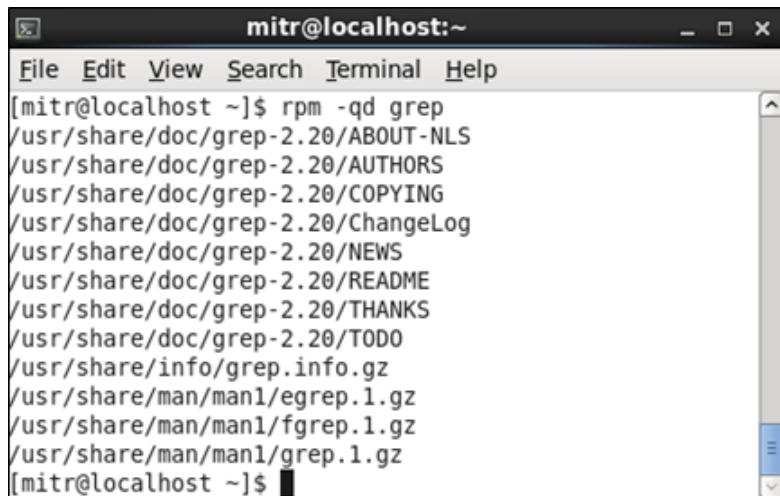
Добавление опции `-v` позволяет расширить информацию о файлах пакета (рис. 11).



```
mitr@localhost:~$ rpm -qc bash
/etc/skel/.bash_logout
/etc/skel/.bash_profile
/etc/skel/.bashrc
mitr@localhost:~$ rpm -qc grep
mitr@localhost:~$
```

Рисунок 12.
Конфигурационные файлы пакетов

Для получения информации только о конфигурационных файлах пакета используются опции `-qc`. Если конфигурационных файлов нет, то не будет выведено ничего. На рис. 12 видно, что у пакета `bash` есть конфигурационные файлы, а у пакета `grep` их нет.



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qd grep  
/usr/share/doc/grep-2.20/ABOUT-NLS  
/usr/share/doc/grep-2.20/AUTHORS  
/usr/share/doc/grep-2.20/COPYING  
/usr/share/doc/grep-2.20/ChangeLog  
/usr/share/doc/grep-2.20/NEWS  
/usr/share/doc/grep-2.20/README  
/usr/share/doc/grep-2.20/THANKS  
/usr/share/doc/grep-2.20/TODO  
/usr/share/info/grep.info.gz  
/usr/share/man/man1/egrep.1.gz  
/usr/share/man/man1/fgrep.1.gz  
/usr/share/man/man1/grep.1.gz  
[mitr@localhost ~]$
```

Рисунок 13.
Файлы документации пакета

Для получения информации только о файлах документации пакета используются опции `-qd` (рис. 13).

Статус	Значение
normal	Файл был установлен
not installed	Файл не был установлен
replaced	Файл был заменен

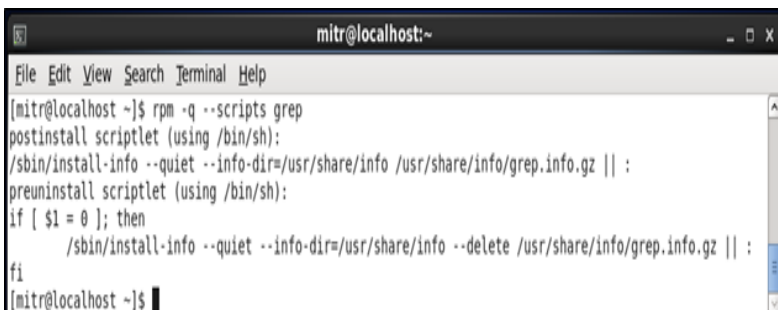
Таблица 3.
Статусы файлов пакета rpm

Опции `-qs` позволяют вывести статус каждого файла пакета. В табл. 3 приведены возможные статусы. Таблица 3. Статусы файлов пакета rpm



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qs grep  
normal      /bin/egrep  
normal      /bin/fgrep  
normal      /bin/grep  
normal      /usr/share/doc/grep-2.20  
normal      /usr/share/doc/grep-2.20/ABOUT-NLS  
normal      /usr/share/doc/grep-2.20/AUTHORS  
normal      /usr/share/doc/grep-2.20/COPYING  
normal      /usr/share/doc/grep-2.20/ChangeLog  
normal      /usr/share/doc/grep-2.20/NEWS  
normal      /usr/share/doc/grep-2.20/README  
normal      /usr/share/doc/grep-2.20/THANKS  
normal      /usr/share/doc/grep-2.20/TODO  
normal      /usr/share/info/grep.info.gz
```

Рисунок 14. Статусы файлов пакета грег



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --scripts grep  
postinstall scriptlet (using /bin/sh):  
/sbin/install-info --quiet --info-dir=/usr/share/info /usr/share/info/grep.info.gz || :  
preuninstall scriptlet (using /bin/sh):  
if [ $1 = 0 ]; then  
    /sbin/install-info --quiet --info-dir=/usr/share/info --delete /usr/share/info/grep.info.gz || :  
fi  
[mitr@localhost ~]$
```

Рисунок 15. Сценарии пакета грег

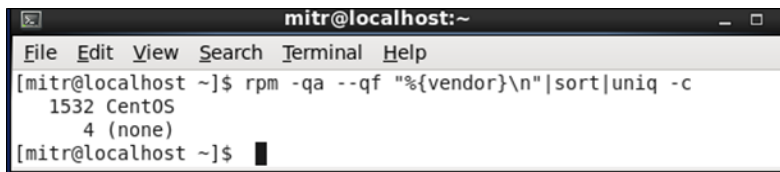
На рис. 14 приводится пример вывода статуса файлов для пакета грег.

Опцию -s можно комбинировать с другими опциями, например с -d для вывода статуса только файлов документации. Пакет грт может содержать сценарии, которые выполняются перед установкой, после установки, перед удалением или после удаления пакета. Опции -q и --scripts позволяют вывести эти сценарии (рис. 15).

```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --changelog grep  
* Tue Sep 15 2015 Jaroslav Škarvada <jskarvad@redhat.com> - 2.20-3.1  
- Made symlinks from egrep, fgrep  
Resolves: rhbz#1263252  
  
* Tue Mar 03 2015 Jaroslav Škarvada <jskarvad@redhat.com> - 2.20-3  
- Updated pcre buildrequires to require pcre-devel >= 7.8-7  
Related: rhbz#1193030  
  
* Mon Feb 16 2015 Jaroslav Škarvada <jskarvad@redhat.com> - 2.20-2  
- Fixed invalid UTF-8 byte sequence error in PCRE mode  
  (by pcre-backported-fixes patch)  
Resolves: rhbz#1193030  
- Fixed buffer overrun for grep -F  
Resolves: CVE-2015-1345  
- Fixed bogus date in the changelog  
  
* Tue Jan 27 2015 Jaroslav Škarvada <jskarvad@redhat.com> - 2.20-1  
- New version  
Resolves: rhbz#1064668  
Resolves: rhbz#982215  
Resolves: rhbz#1126757  
Resolves: rhbz#1167766  
Resolves: rhbz#1171806  
- Fixed \w and \W behaviour in multibyte locales  
Resolves: rhbz#799863  
- Documented --fixed-regexp option  
Resolves: rhbz#1103270  
  
* Wed Mar 26 2014 Jaroslav Škarvada <jskarvad@redhat.com> - 2.6.3-6  
- Fixed UTF-8 in PCRE (pcre-utf-8-fix patch)  
Resolves: rhbz#683753  
  
* Mon Mar 17 2014 Jaroslav Škarvada <jskarvad@redhat.com> - 2.6.3-5  
- Fixed --include (include-fix patch)  
Resolves: rhbz#1040710
```

Опции `-q` и `--changelog` позволяет вывести список изменений, произошедших в пакете, по сравнению с прошлыми версиями. На рис. 16 представлен сокращенный список изменений пакета `grep`.

Рисунок 16.
Список изменений пакета `grep`



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -qa --qf "%{vendor}\n"|sort|uniq -c  
1532 CentOS  
4 (none)  
[mitr@localhost ~]$
```

Рисунок 17.

Вывод информации о поставщиках пакетов



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "[%{name} - %{filenames} - %{filesizes}\n]" grep  
grep - /bin/egrep - 4  
grep - /bin/fgrep - 4  
grep - /bin/grep - 155472  
grep - /usr/share/doc/grep-2.20 - 4096  
grep - /usr/share/doc/grep-2.20/ABOUT-NLS - 93787  
grep - /usr/share/doc/grep-2.20/AUTHORS - 2309  
grep - /usr/share/doc/grep-2.20/COPYING - 35148  
grep - /usr/share/doc/grep-2.20/ChangeLog - 275436  
grep - /usr/share/doc/grep-2.20/NEWS - 30246  
grep - /usr/share/doc/grep-2.20/README - 2248  
grep - /usr/share/doc/grep-2.20/THANKS - 5272  
grep - /usr/share/doc/grep-2.20/TODO - 11023
```

Рисунок 18.

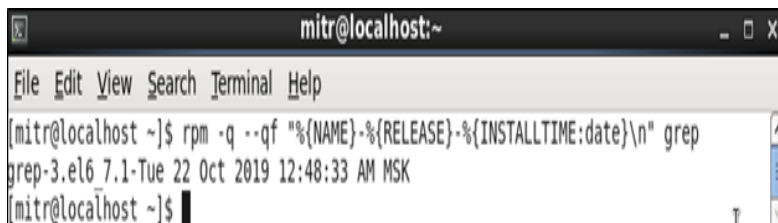
Вывод элементов заголовка,
являющихся массивами

Опция `--qf` или `--queryformat` позволяет создавать собственные запросы. Для создания запроса опции `--qf` необходимо передать форматную строку, синтаксис которой сходен с функцией `printf` языка программирования C. Базовый синтаксис формата запроса: `%{tag_name}`. Напомним, что имена тегов могут быть получены с помощью команды `rpm --querytags`. Можно комбинировать имена полей (тегов) для вывода более чем одного значения поля для указанного пакета. Также можно добавлять опции форматирования в соответствии с соглашениями языка программирования C.

Например, с помощью команды, приведенной на рис. 17 можно определить источники установленных в системе пакетов.

В нашем случае из 1536 пакетов, установленных в системе, 1532 поставлены CentOS, поставщики 4 пакетов неизвестны. Некоторые записи заголовка представляют собой массивы, то есть содержат более чем одно значение. Для запроса каждой записи в массиве используются квадратные скобки.

В примере на рис. 18 выводятся имя пакета, которому принадлежит файл, файлы, входящие в пакет, и их размеры, для пакета `grep`. Символ «`=`» использован перед `name` для того, чтобы показать, что `name` – не массив, а значение одно и то же значение `name` необходимо выводить перед каждой строкой вывода элементов массивов `filenames` и `filesizes`.



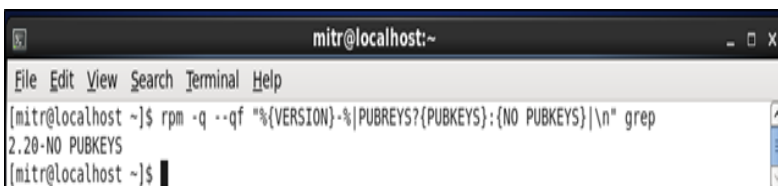
```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "%{NAME}-%{RELEASE}-%{INSTALLTIME:date}\n" grep  
grep-3.el6 7.1-Tue 22 Oct 2019 12:48:33 AM MSK  
[mitr@localhost ~]$
```

Рисунок 19.



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "%{NAME}-%{RELEASE}-%{INSTALLTIME}\n" grep  
grep-3.el6 7.1-1571694513  
[mitr@localhost ~]$
```

Рисунок 20.



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "%{VERSION}-%{PUBKEYS?{PUBKEYS}:{NO PUBKEYS}}\n" grep  
2.20-NO PUBKEYS  
[mitr@localhost ~]$
```

Рисунок 21.

Некоторые поля заголовка содержат информацию в виде, в котором которую их не имеет смысла выводить. Для обработки подобных случаев используется синтаксис `%{tag:special_format}`.

Так для вывода времени инсталляции пакета, которое хранится в поле `INSTALLTIME`, применяется `%{INSTALLTIME:date}`. В примере (рис. 19) выводятся имя, релиз и дата инсталляции пакета `grep`.

Для сравнения на рис. 20 приведен вывод команды, не использующей специальный формат «:date».

Большая часть тегов в заголовке не является обязательной, то есть в конкретном пакете их может и не быть. Если тег в пакете не определен, то команда обращения к нему ничего не возвращает. Для обработки такого случая можно использовать следующий условный оператор (рис. 21): `%{tag?{выводить если есть}:{выводить если нет}}`

Ряд тегов содержит информацию о зависимостях пакета. Каждый из этих тегов существует в рамках троек тегов, которые форматируются сходным образом.

Например, для описания зависимостей, которые требуется пакету, используются поля REQUIRENAME, REQUIREVERSION и REQUIREFLAGS. REQUIRENAME хранит имя требуемой зависимости, REQUIREVERSION - массив подходящих версий, REQUIREFLAGS объединяет значения двух предыдущих полей с использованием битовых флагов, которые задают статус зависимости в терминах «имеющаяся версия больше нужной», «имеющаяся версия равна нужной», «имеющаяся версия меньше нужной». В табл. 3 перечислены теги зависимостей.


Категория	Тег	Поле содержит
С чем пакет конфликтует	CONFLICTFLAGS	Массив флагов для конфликтов
	CONFLICTFLAGS	Массив имен, с которыми пакет конфликтует
	CONFLICTFLAGS	Массив номеров версий, с которыми пакет конфликтует
От чего пакет зависит	REQUIREFLAGS	Массив флагов зависимостей
	REQUIREFLAGS	Массив имен пакетов, в которых пакет нуждается
	REQUIREFLAGS	Массив номеров версий пакетов, в которых пакет нуждается
Не актуально (устарело)	OBSOLETENAME	Массив имен, которые пакет делает неактуальными
	OBSOLETENAME	Массив флагов не актуальных пакетов
	OBSOLETENAME	Массив номеров версий которые пакет делает неактуальными
Что пакет предоставляет	PROVIDENAME	Массив имен возможностей, которые пакет предоставляет
	PROVIDENAME	Массив флагов возможностей, которые пакет предоставляет
	PROVIDENAME	Массив номеров версий возможностей, которые пакет предоставляет

Таблица 3.
Теги зависимостей



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "%{REQUIRENAME}%{REQUIREFLAGS:depflags}%{REQUIREVERSION}\n" grep  
/bin/sh**  
/sbin/install-info**  
/sbin/install-info**  
libc.so.6**  
libc.so.6(GLIBC_2.0)**  
libc.so.6(GLIBC_2.1)**  
libc.so.6(GLIBC_2.1.1)**  
libc.so.6(GLIBC_2.1.3)**  
libc.so.6(GLIBC_2.2)**  
libc.so.6(GLIBC_2.3)**  
libc.so.6(GLIBC_2.3.4)**  
libc.so.6(GLIBC_2.4)**  
libpcre.so.0**  
rpmLib(CompressedFileNames)*<=3.0.4-1  
rpmLib(FileDigests)*<=4.6.0-1  
rpmLib(PayloadFilesHavePrefix)*<=4.0-1  
rtld(GNU_HASH)**  
rpmLib(PayloadIsXz)*<=5.2-1  
[mitr@localhost ~]$
```

Рисунок 22. Зависимости пакета грег



```
mitr@localhost:~  
File Edit View Search Terminal Help  
[mitr@localhost ~]$ rpm -q --qf "%{CONFLICTNAME}%{CONFLICTFLAGS:depflags}%{CONFLICTVERSION}\n" grep  
[mitr@localhost ~]$ rpm -q --qf "%{PROVIDENAME}%{PROVIDEFLAGS:depflags}%{PROVIDEVERSION}\n" grep  
bundled(gnulib)**  
grep*2.20-3.el6_7.1  
grep(x86-32)*2.20-3.el6_7.1  
[mitr@localhost ~]$
```

Рисунок 23.
Конфликты и удовлетворяемые
зависимости пакета грег

Для их просмотра массива флагов удобно пользоваться специальным форматом флагов «:depflags». На рис. 22 показано, как получить зависимости для пакета грег.

Из рис. 22 видно, что грег зависит от пакетов, вывод которых завершается «**» независимо от их версии. Версии пакетов, после имен которых следует «*<=*», не должны превышать версий, указанных после этой комбинации символов. Заметим, что символ «*» является частью формата вывода, который мы определили сами. Рис. 23 показывает, что пакет грег ни с кем не конфликтует и предоставляет средства для удовлетворения указанных на рис. 23 зависимостей.

Основы системного администрирования и сетевых технологий
