

Пакеты

Способ упаковки программ в Linux называется пакет. Ubuntu использует deb-пакеты, в других репозиториях может использоваться такой же вид пакетов либо другие (например, rpm). Так как у нас ubuntu, далее мы будем рассматривать именно deb-пакеты.

Технически **deb-пакет** — это архив. Он состоит из самой программы и различных файлов со служебной информацией — о том, как установить программу, куда, что делать перед установкой и что делать в случае обновления или удаления пакета; и с информацией о самом пакете — кто его создатель, какая версия и прочее.

Аналог Apple Store в Linux называется **репозиторием (repo)**. Как и другие программы, репозиторий распространяется в виде пакета (или нескольких зависимых друг от друга пакетов) и этот пакет хранится в репозитории.

Репозиторий — это технически набор директорий, организованных специальным образом. Работать с репозиториями удобнее всего при помощи специальных программ.

Репозиторий — это внешняя база данных, по отношению к операционной системе. Но есть и внутренняя, сама операционная система хранит информацию о том, какие пакеты были установлены и какие могут быть установлены. Обратите внимание, что интерактивного поиска в репозитории нет. Поэтому операционная система скачивает из репозитория специально подготовленный файл с информацией о том, какие пакеты и каких версий можно скачать из соответствующего репозитория. И этот файл тоже который хранится в репозитории.

Работа с deb-пакетами

В Linux есть несколько программ для работы с deb-пакетами.

Внутренняя база: dpkg (читается дэпэкэдж), apt-cache, apt.

Внешняя: apt-get, apt.

dpkg не умеет связываться с репозиториями, но позволяет работать с пакетами, которые уже установлены, или установить уже скачанный deb-пакет.

Случай, когда вы не скачиваете пакет из репозитория, а скачиваете файл со стороннего сайта (он имел бы название something.deb), относительно редкий. Чаще всего это случается с какими-то специализированными программами или когда вы собрали пакет сами и хотите установить его, например чтобы проверить, что с ним всё в порядке.

Рассмотрим основные случаи использования dpkg.

Узнать, какие пакеты установлены на машине:

```
dpkg -l
```

Узнать, кто мейнтейнер пакета (так называется человек, который занимается превращением программы в пакет и занимается поддержкой и обновлением пакета) и расширенную информацию о пакете:

```
dpkg -s telegram-desktop
```

Узнать, какие файлы в операционной системе принадлежат определённому пакету:

```
dpkg -L telegram-desktop
```

Если бы мы скачали `program.deb`, мы могли бы установить его при помощи:

```
dpkg -i program.deb
```

Чтобы потом этот пакет удалить:

```
dpkg -r program.deb
```

Расскажет, что ещё умеет программа:

```
dpkg -help
```

apt-get — это программа, которой мы будем пользоваться чаще всего.

Чтобы обновить эту информацию (например, чтобы узнать, для каких пакетов появились новые версии), используется программа:

```
sudo apt-get update
```

Так как эта команда меняет содержимое служебных файлов (скачивает нам новый кэш пакетов), нам понадобится *sudo*.

```
apt-get install telegram-desktop
```

Если пакет уже установлен, но существует более новая версия, этот пакет обновится. Если вы хотите установить несколько пакетов, вы можете перечислить их через пробел.

```
apt-get remove -purge telegram-desktop
```

Эта команда удалит пакет и все его файлы, в том числе конфигурацию.

```
apt-get changelog telegram-desktop
```

Эта команда покажет чейнджлог. Лог изменений. Так называется информация о том, что менялось в различных версиях.

```
apt-get upgrade
```

Эта команда обновит все установленные пакеты.

Команда *apt-cache* умеет работать с внутренней базой данных — кэшем.

Узнать, какие нам доступны пакеты:

```
apt-cache pkgnames
```

Используйте осторожно, список будет длинным.

Узнать название пакета, зная название программы:

```
apt-cache search telegram
```

Для Telegram существует не только тот пакет, который мы установили (с графическим интерфейсом), но и пакет с консольной версией. Вы можете установить его и попробовать воспользоваться телеграмом прямо из консоли.

Представим, что в последней версии клиента телеграма отключили стикеры. Для этого давайте посмотрим, какие есть версии телеграма в репозитории:

```
apt-cache policy telegram-desktop
```

Кроме того, обратите внимание, что эта же команда покажет нам, установлен ли пакет в системе.

Установить какую-то другую версию:

```
apt-get install telegram-desktop=xxx
```

Но как *apt-get* узнает, откуда ему нужно скачать пакет? Информация для этого хранится в файле */etc/apt/sources.list*

Кроме программы *apt-get*, существует программа **apt**.

Это пользовательский интерфейс к программе *apt-get*. У неё есть свои плюсы, но они в основном связаны с пользовательским опытом — там по умолчанию включена подсветка цветом, прогресс-бар процесса установки, она объединяет в себе функции *apt-get*, *apt-cache* и *dpkg*.

Давайте установим пакет *sl*.

```
apt-get install sl
```

Вы помните, что, для того чтобы посмотреть, какие файлы есть в директории (и узнать информацию о них), мы пользовались командой *ls*. Теперь у нас есть программа, которая, если мы опечатаемся в этой команде и наберём *sl*, покажет нам поезд.