

Установка ПО (бинарные пакеты и исходники), работа с архивами

Цель работы: изучение технологии установки прикладного программного обеспечения и технологии работы с архивами

Как правило, в Windows-системах установка программного обеспечения (ПО) осуществляется с помощью специальных программ (обычно их называют `setup.exe` или `install.exe`), хотя в последнее время всё чаще ПО устанавливают с помощью специально созданных MSI-файлов. Такие файлы содержат установочный пакет для быстрой и безупречной установки на платформе Windows. MSI-файл может быть использован как сторонними разработчиками для установки программного обеспечения, так и для обновления Windows. Эти файлы содержат всю информацию, которая необходима установщику Windows для установки или удаления приложения или продукта, и для конфигурации пользовательского интерфейса.

В Linux-системах установка ПО осуществляется несколько иначе, но тоже с использованием пакетов. Главное, что дистрибутивы программ (сама программа и вспомогательные файлы — файлы справки, конфигурационные файлы) поставляются именно в виде пакетов.

Установка ПО в Linux может быть выполнена из бинарных пакетов (наподобие виндовых `setup.exe`) и из исходников. Двоичный формат может различаться для разных дистрибутивов и версий ОС. Это самый простой и самый распространённый способ.

Пакеты, которые могут быть установлены в систему, хранятся в так называемых репозиториях. Репозитории – это специальные серверы-хранилища таких файлов. Их также можно назвать источниками приложений. Пользовательские компьютеры (и серверы) подключаются к репозиториям по сети или через интернет и при помощи специальных утилит (таких, например, как `Synaptic` или `software-center` в Ubuntu Linux) позволяют увидеть, какие пакеты у нас установлены, какие доступны для установки. Большинство утилит поддерживают простой поиск по ключевым словам и способны разбивать группы пакетов по категориям.

Использование связки «репозиторий + система управления пакетами» позволяет использовать простой, централизованный метод установки/удаления программ, а также предоставляет удобный способ выкладывания обновлений. Различают два основных формата пакетов: для дистрибутивов семейства Debian и для дистрибутивов семейства RedHat.

Установка пакетов в системах семейства Debian

Для дистрибутивов семейства Debian, к которому относится Ubuntu Linux, бинарники пакетов представлены в виде файлов с расширением `*.deb` и установка некоего пакета может быть сделана с помощью программы **`dpkg`**. **`dpkg`** — это системное программное обеспечение, пакетный менеджер. Она может устанавливать, удалять и создавать пакеты, но,

в отличие от других систем управления пакетами, она не может автоматически загружать и устанавливать пакеты или их зависимости.

Например, чтобы получить список пакетов, установленных в системе, в терминале наберите:

```
sudo dpkg -l
```

А чтобы установить пакет **file.deb** можно воспользоваться командой

```
sudo dpkg -i file.deb
```

Удаление пакета может быть выполнено по команде

```
sudo dpkg -r file.deb
```

Помимо низкоуровневого пакетного менеджера **dpkg** в Ubuntu имеется высокоуровневые средства. При работе из командной строки это команда (утилита) **apt** - это мощный консольный инструмент, который работает с Улучшенным инструментарием пакетов (APT — Advanced Package Tool) , выполняющий такие функции, как установка новых программных пакетов, обновление имеющихся пакетов, обновления индекса списка пакетов и даже обновление все системы Ubuntu.

Будучи простым консольным инструментом, **apt** имеет ряд преимуществ над другими инструментами управления пакетами, доступными в Ubuntu. Некоторые из этих преимуществ включают легкое использование простых терминальных соединений (SSH) а также есть возможность использования в сценариях системных администраторов, которые могут быть автоматизированы с помощью утилиты планирования **cron**.

Некоторые примеры использования утилиты **apt**:

Установка пакетов с использованием инструмента **apt** очень проста. Например, для установки пакета ПО для работы с RAID-массивами - mdadm - наберите следующее:

```
sudo apt install mdadm
```

Для удаления пакета, установленного в предыдущем примере, наберите следующее:

```
sudo apt remove mdadm
```

Установка пакетов в системах семейства RedHat

Для Linux-систем семейства Red Hat бинарники — это файлы с расширением *.rpm . В качестве менеджера управления пакетами выступает одноимённая программа **rpm**. Формат команды для установки пакета **file.rpm** будет следующим:

```
rpm -ihv file.rpm
```

Если Вы установили себе систему Linux, которая использует репозитории с rpm-пакетами, то почитайте документацию и установите с помощью менеджера пакетов какую-нибудь программу. Само собой, что условный пакет **file.rpm** у Вас должен быть «под рукой».

Например, в качестве устанавливаемой программы может быть какой-нибудь текстовый редактор, который не был установлен в Вашу систему.

Помимо низкоуровневого пакетного менеджера **rpm** в Linux-системах, которые следуют стандарту LSB (Linux Standard Base) часто используют высокоуровневый консольный менеджер пакетов **yum** и его «расширения» с графическим интерфейсом. Выполните команду **yum help** и почитайте как можно получить список установленных пакетов, как можно получить перечень возможных для установки пакетов и про другие возможности.

Архивация

Термин архивация означает создание так называемого архива, т. е. копии файлов, которой можно воспользоваться для восстановления данных в случае какой-нибудь аварии. А для того, чтобы архивы занимали меньше места на носителях, их дополнительно сжимают. Но в чистом виде различают процедуру архивирования и процедуру сжатия.

Основным средством архивирования в POSIX-системах (а, следовательно, и в Linux) является комплекс из двух программ — **tar** и **gzip**. Хотя никто не запрещает пользоваться **arj**, **pkzip**, **lha**, **rar** и т. д. — версии этих программ для Linux общедоступны. Просто исторически уже сложилось, что пользователи Unix чаще применяют именно **tar** и **gzip**, и именно в таком формате распространяется большая часть программного обеспечения для Linux. Поэтому желательно овладеть работой с **tar** и **gzip**.

tar расшифровывается как Tape ARchiver, он не сжимает данные, а лишь объединяет их в единый файл с последовательным доступом для последующей записи на ленту. В этом файле помимо данных, попавших в архив, сохраняется информация о структуре каталогов, в которых располагались исходные файлы, попавшие в архив. Изначально архивные файлы создавались на ленточных накопителях, точнее на устройстве **/dev/rmt0**. Поэтому если мы хотим создать архивный файл на диске, то необходимо использовать команду **tar** с опцией **f**, после которой указывается имя архивного файла.

Чтобы создать один **tar**-архив из нескольких файлов, используется команда:

```
[user]$ tar -cf имя_архива файл1 файл2 ...,
```

где опция **-c** сообщает программе, что необходимо создать (create) архив, а опция **f** говорит о том, что архив должен создаваться в виде файла (имя которого должно следовать сразу за этой опцией).

В именах файлов, которые сохраняются в архиве, можно использовать шаблоны имен файлов, в том числе просто символы-заместители ***** и **?**. Благодаря этому можно очень короткой командой отправить в архив сразу много файлов. Например, для того, чтобы создать архив, содержащий все файлы одного из подкаталогов (пусть это будет **subdir**) текущего каталога, достаточно дать команду

```
[user]$ tar -cvf имя_архива ./subdir/*
```

или даже просто `[user]$ tar -cvf имя_архива subdir`

По этой команде в архиве будут сохранены не только файлы, расположенные непосредственно в подкаталоге **subdir**, но и рекурсивно все файлы из подкаталогов каталога **subdir**. При этом в архиве сохраняется вся структура подкаталогов каталога **subdir**.

Установка ПО из исходников

Поскольку существует огромное множество дистрибутивов Linux, сделать бинарные пакеты под все их вариаций представляется достаточно сложной задачей. Поэтому, зачастую ПО распространяется в виде архива с исходными текстами программы, требующими компиляции перед установкой. Это позволяет сделать их по-настоящему переносимыми. Пакеты с исходным кодом **содержат код и сценарии для сборки приложений и библиотек**. Кроме того, в пакете могут содержаться патчи к исходному коду, управляющие скрипты, файлы ресурсов приложений, документация. Обычно одному пакету с исходниками соответствует один бинарный пакет, хотя это и не всегда так. Обычно установка программы из исходников сводится к выполнению нескольких простых действий:

1) Скачайте пакет (часто они могут быть представлены как archive.tar.gz)

2) Распакуйте архив при помощи команд:

```
gzip -d archive.tar.gz
```

```
tar -xvf archive.tar
```

3) Перейдите в каталог с распакованными файлами

4) Прочитайте файл README.TXT (этот шаг лучше не пропускать, хотя частенько люди нарушают эту рекомендацию)

5) Сконфигурируйте приложение, запустив конфигурационный скрипт:

```
./configure
```

6) Скомпилируйте ПО при помощи команды:

```
make
```

7) Установите исполняемые файлы, выполнив команду (от имени суперпользователя):

```
sudo make install
```

Если на одном из шагов возникли ошибки - придется внимательно читать пропущенный README.TXT - там обычно есть рекомендации по установке. Заодно почитайте про команду **make**.

Неплохое руководство (далеко не единственное) по установке ПО из исходных кодов можно посмотреть здесь - https://help.ubuntu.ru/wiki/programs_installation

Задание:

1. Установите бинарный пакет для утилиты mdadm, которая позволяет работать с RAID-массивами дисков. Прочитайте про работу с ней, попробуйте ее в работе, а затем удалите. Для работы с ней желательно в компьютер добавить еще один виртуальный диск, а лучше даже два.

2. А теперь скачайте пакет исходников и установите одно из неустановленных у вас приложений (например, это может быть Midnight Commander, Double Commander, GIMP или др.). Главное — чтобы пакет был не бинарный, а из исходников. Один из перечней возможных программ можете посмотреть например здесь: <http://itshaman.ru/it-programmy-dlya-linux/> . Опишите в отчете возникшие сложности и как Вы их решили.