

Лабораторная работа № 4

Установка программного обеспечения

Задачи:

1. Научиться получать информацию об установленных и доступных пакетах и устанавливать пакеты программного обеспечения из репозитория в Linux.
2. Научиться устанавливать программные продукты из исходных кодов в Linux.
3. Научится конфигурировать менеджер пакетов и создавать собственные репозитории.

Рекомендации:

1. Программные продукты под Linux распространяются или в виде готовых бинарных пакетов (с расширением deb для ветви Debian и расширением rpm для ветви RedHat) или в виде исходных кодов.
2. Из бинарного пакета программное обеспечение устанавливается с помощью специального ПО — менеджера пакетов (например, dpkg и apt для Debian ветви и rpm или dnf (в ранних версиях yum) для ветви RedHat). Менеджеры пакетов также позволяют работать с репозиториями — коллекциями пакетов, некоторые из которых поддерживаются производителями дистрибутивов. На работу с такими репозиториями, как правило, по умолчанию настроена система.
3. Для установки, обновления и просмотра информации о пакетах из сетевого репозитория используется менеджер пакетов **dnf**.
4. Исходный код программ, в большинстве случаев, распространяется в архивах. Для создания этих архивов, как правило, используются две программы: **tar** и **gzip**. Первая собирает все исходные файлы, включая структуру каталогов в один общий файл, а вторая сжимает этот файл для того, чтобы сэкономить время передачи архива по сети.
5. Сборка и установка пакетов из исходных кодов осуществляется с помощью последовательного запуска команд:
make
make install
Эти команды должны быть запущены так, чтобы текущим каталогом при этом был каталог с исходным кодом устанавливаемого пакета, содержащий файл с именем **Makefile**.
6. Для создания собственного репозитория должен быть установлен и использован пакет **createrepo**.
7. Для каждого репозитория в системе должен быть конфигурационный файл. Эти файлы располагаются в каталоге: **/etc/yum.repos.d**. Файл обязан иметь расширение **.repo**. Файл имеет следующую структуру (курсивом отмечены изменяемые значения):
[название_репозитория]
name=название_репозитория
mirrorlist=список_адресов_репозитория
enabled=значение_флага_доступности
gpgcheck=значение_флага_проверки_цифровой_подписи
Примечания: Если репозиторий находится в локальном доступе, то в качестве адреса указывается:
file://полный_путь_к_каталогу/
Если пакеты получены из надежных источников, значение флага проверки цифровой подписи можно поставить равное 0 и не настраивать проверку ключей.
8. В случае необходимости установить приложение, существующее в виде пакета Debian, необходимо воспользоваться утилитой **alien**. Если её нет в стандартном репозитории, можно найти репозиторий, содержащий эту утилиту или установить её из исходного кода.

Задание:

Выполните указанные ниже действия и запишите все использовавшиеся команды с параметрами и изменения в файлах в отдельный файл:

1. Установите из сетевого репозитория пакеты, входящие в группу «Development Tools».
2. Установите из исходных кодов, приложенных к методическим указаниям пакет **bastet-0.43**. Для этого необходимо создать отдельный каталог и скопировать в него исходные коды проекта **bastet**. Вы можете использовать подключение сетевого каталога в хостовой операционной системе для передачи архива с исходными кодами в виртуальную машину. Далее следует распаковать архив до появления каталога с исходными файлами (в каталоге должен отображаться **Makefile**). После этого соберите пакет **bastet** и запустите его, чтобы удостовериться, что он правильно собрался. Затем модифицируйте **Makefile**, добавив в него раздел **install**. Обеспечьте при установке копирование исполняемого файла в

/usr/bin с установкой соответствующих прав доступа. Выполните установку и проверьте, что любой пользователь может запустить установленный пакет.

3. Создайте файл **task3.log**, в который выведите список всех установленных пакетов.
4. Создайте файл **task4_1.log**, в который выведите список всех пакетов (зависимостей), необходимых для установки и работы компилятора **gcc**. Создайте файл **task4_2.log**, в который выведите список всех пакетов (зависимостей), установка которых требует установленного пакета **libgcc**.
5. Создайте каталог **localrepo** в домашнем каталоге пользователя root и скопируйте в него пакет **checkinstall-1.6.2-3.el6.1.x86_64.rpm**, приложенный к методическим указаниям. Создайте собственный локальный репозиторий с именем localrepo из получившегося каталога с пакетом.
6. Создайте файл **task6.log**, в который выведите список всех доступных репозиторий.
7. Настройте систему на работу только с созданным локальным репозиторием (достаточно переименовать конфигурационные файлы других репозиторий). Выведите на экран список доступных для установки пакетов и убедитесь, что доступен только один пакет, находящийся в локальном репозитории. Установите этот пакет.
8. Скопируйте в домашний каталог пакет **fortunes-ru_1.52-2_all**, приложенный к методическим рекомендациям, преобразуйте его в rpm пакет и установите.
9. Скачайте из сетевого репозитория пакет **nano**. Пересоберите пакет таким образом, чтобы после его установки менеджером пакетов, появлялась возможность запустить редактор **nano** из любого каталога, введя команду **newnano**.
10. Предъявите преподавателю изменения в системе и файл с описанием использованных команд.