

Лабораторная работа 17-18

Установка и обновление прикладного ПО в Ubuntu в отсутствии интернета через локальные репозитории и сигнатуры (самостоятельная работа)

Ход работы:

Использование локальных репозиториев:

1. Создайте через терминал в корневом каталоге папку с именем “repos”

```
acuioc-ubuntu@acuioc-ubuntu-1-2:~$ sudo mkdir /repos  
[sudo] password for acuioc-ubuntu:
```

2. Зайдите через терминал в папку /etc/apt

```
acuioc-ubuntu@acuioc-ubuntu-1-2:~$ cd /etc/apt
```

3. Включите режим полного доступа к этой папке

```
acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ sudo chmod 777 /etc/apt
```

4. Создайте файл sources.list.d/local.list

```
acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ sudo touch sources.list.d
```

5. Командой nano откройте файл и впишите в него локальный репозиторий deb file:///путь до файла *.deb

```
acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ sudo nano sources.list.d  
acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$
```

```
GNU nano 7.2 sources.list.d  
deb file:/// *.deb./
```

6. Сохраните. Теперь вы можете использовать папку как собственный репозиторий, по аналогии со скриптом в прошлой Лабораторной работе, для это достаточно поместить в указанную папку дистрибутивы *.deb и использовать команду `sudo apt-get install имя пакета (или всех пакетов в папке *.deb)`

7. При установки дистрибутивов, они могут иметь зависимости, для проверки зависимостей необходимо ввести команду

`apt show`

имя_пакета

например:

```
acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ sudo apt show mozilla
Package: firefox
Version: 1:1snap1-0ubuntu3
Priority: optional
Section: web
Origin: Ubuntu
Maintainer: Ubuntu Mozilla Team <ubuntu-mozillateam@lists.ubuntu.com>
Bugs: https://bugs.launchpad.net/ubuntu/+filebug
Installed-Size: 135 kB
Provides: gnome-www-browser, iceweasel, www-browser, x-www-browser
Pre-Depends: debconf, snapd (>= 2.54)
Depends: debconf (>= 0.5) | debconf-2.0
Breaks: firefox-dbg (<< 1:1snap1), firefox-dev (<< 1:1snap1),
        r (<< 1:1snap1), firefox-mozsymbols (<< 1:1snap1)
Replaces: firefox-dbg (<< 1:1snap1), firefox-dev (<< 1:1snap1),
        ver (<< 1:1snap1), firefox-mozsymbols (<< 1:1snap1)
Task: ubuntu-desktop-minimal, ubuntu-desktop, xubuntu-desktop,
      op, ubuntu-unity-live, edubuntu-desktop-gnome, ubuntucinnamon-
      namon-desktop-raspi
Download-Size: 71.9 kB
APT-Manual-Installed: no
APT-Sources: http://archive.ubuntu.com/ubuntu mantic/main amd64
Description: Transitional package - firefox -> firefox snap
     This is a transitional dummy package. It can safely be removed.
```

```
Provides: gnome-www-browser, iceweasel, www-browser, x-www-browser
Pre-Depends: debconf, snapd (>= 2.54)
Depends: debconf (>= 0.5) | debconf-2.0
Breaks: firefox-dbg (<< 1:1snap1), firefox-dev (<< 1:1snap1),
        r (<< 1:1snap1), firefox-mozsymbols (<< 1:1snap1)
Replaces: firefox-dbg (<< 1:1snap1), firefox-dev (<< 1:1snap1),
        ver (<< 1:1snap1), firefox-mozsymbols (<< 1:1snap1)
Task: ubuntu-desktop-minimal, ubuntu-desktop, xubuntu-desktop,
      op, ubuntu-unity-live, edubuntu-desktop-gnome, ubuntucinnamon-
      namon-desktop-raspi
Download-Size: 71.9 kB
APT-Manual-Installed: no
APT-Sources: http://archive.ubuntu.com/ubuntu mantic/main amd64
Description: Transitional package - firefox -> firefox snap
     This is a transitional dummy package. It can safely be removed.
     .
     firefox is now replaced by the firefox snap.
```

```
N: Unable to locate package mozilla
N: Unable to locate package mozilla
```

```

acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ apt-cache depends m
firefox
  PreDepends: debconf
  PreDepends: snapd
|Depends: debconf
  Depends: <debconf-2.0>
    cdebconf
    debconf
  Breaks: <firefox-dbg>
  Breaks: <firefox-dev>
  Breaks: <firefox-geckodriver>
  Breaks: <firefox-mozsymbols>
  Replaces: <firefox-dbg>
  Replaces: <firefox-dev>
  Replaces: <firefox-geckodriver>
  Replaces: <firefox-mozsymbols>

```

8. Проверьте зависимости любого ПО, установленного в Ubuntu
9. Также, сторонние пакеты могут иметь зависимости и при удалении пакетов, зависимости могут остаться в системе, для очистки зависимостей используются команды

```
sudo apt autoclean
```

```

acuioc-ubuntu@acuioc-ubuntu-1-2:/etc/apt$ sudo apt autoclean
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done

```

```
sudo apt clean
```

```
sudo apt autoremove
```




10. Введите поочередно эти команды и посмотрите, что будет происходить

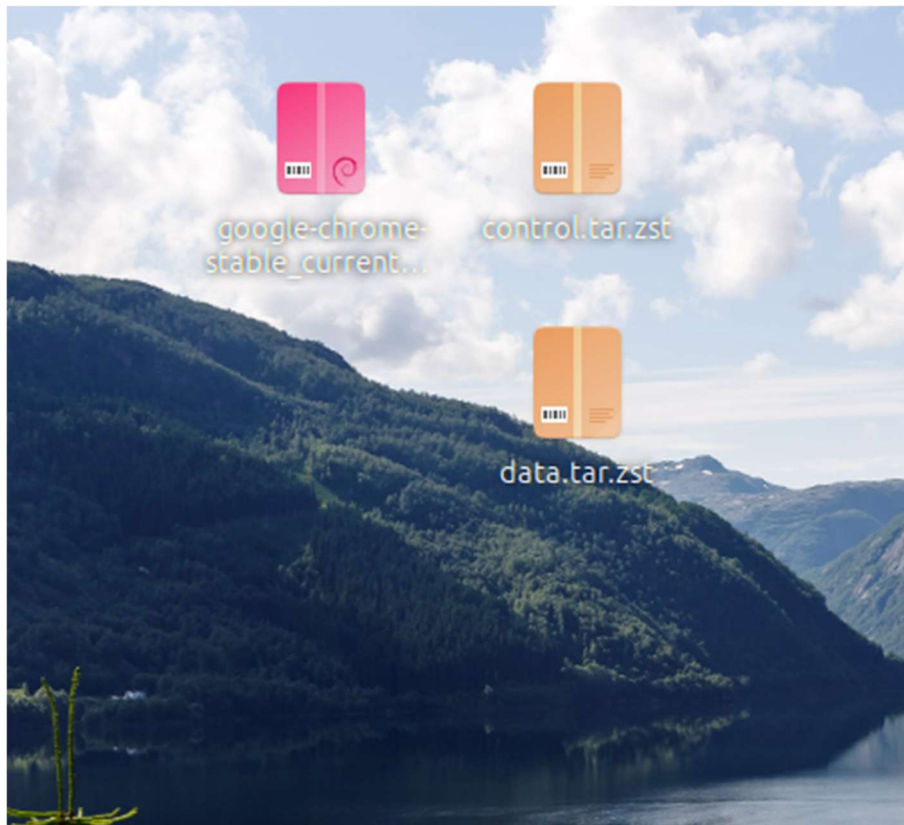
Полное обновление файлов в Ubuntu в оффлайн-режиме

1. Установите Apt-offline в автономной системе:

Ссылка на дистрибутив:

http://archive.ubuntu.com/ubuntu/pool/universe/a/apt-offline/apt-offline_1.8.5-1_all.deb

Имя	Дата изменения	Тип	Размер
 control.tar.zst	11.08.2022 20:25	Файл "ZST"	3 КБ
 data.tar.zst	11.08.2022 20:25	Файл "ZST"	49 КБ
 google-chrome-stable_current_amd64.deb	03.04.2025 9:01	deb Archive	112 474 КБ



2. Создайте через терминал в корневом каталоге папку с именем “apt-offline-dir”
3. Перейдите в нее и создайте файлы сигнатуры:

```
acuioc-ubuntu@acuioc-ubuntu-1-2:~$ cd /apt-offline-dir
acuioc-ubuntu@acuioc-ubuntu-1-2:/apt-offline-dir$
```

`sudo apt-offline set apt-offline.sig --update --upgrade`

Пример вывода будет таким:

*Generating database of files that are needed for an update.
Generating database of file that are needed for operation
upgrade*

По умолчанию apt-offline генерирует базу данных файлов, которые необходимо обновить.

Опции `--update` или `--upgrade` нужны для создания базы данных.

4. Скопируйте всю папку apt-offline-dir на USB-накопитель или внешний накопитель и перейдите в Ubuntu с онлайн-системой (т. е. систему с интернетом).

Шаги, которые нужно выполнить в онлайн системе

1. Подключите USB-накопитель и перейдите через терминал в каталог «apt-offline-dir»
:
2. Затем выполните следующую команду:

```
$ sudo apt-offline get apt-offline.sig --threads 5 --bundle apt-offline-bundle.zip
```

Тут «-threads 5» представляет количество репозиториев АРТ.

Вы можете увеличить это количество, если хотите загрузить пакеты из большего количества репозиториев.

А опция «--bundle apt-offline-bundle.zip» представляет все пакеты, которые будут объединены в один архивный файл с именем apt-offline-bundle.zip.

Этот архивный файл будет сохранен в вашем текущем рабочем каталоге.

Приведенная выше команда загрузит данные на основе файла сигнатуры, созданного ранее в автономной системе.

Это займет несколько минут в зависимости от скорости интернет-соединения.

Apt-offline является кроссплатформенным, поэтому вы можете использовать его для загрузки пакетов в любой ОС (можете поэкспериментировать на системах Windows).

3. После завершения вернитесь обратно в автономную систему с USB-накопителем и созданным архивом обновлений.

Действия в автономной системе

1. Подключите устройство к автономной системе и перейдите в каталог apt-offline-dir, куда вы скачали все пакеты ранее.

2. Затем выполните следующую команду, чтобы установить все загружаемые пакеты.

```
$ sudo apt-offline install apt-offline-bundle.zip
```

3. Это обновит и заполнит базу данных АРТ загруженными пакетами в вашей автономной системе.

Comands conspect:

Sudo su

Exit

Cp

Ls

Cd

Address: local / absolut
Sh / bash – processing

Text: Cat – show me text of file

Ответы на вопросы:

1. **Для чего нужен Репозиторий?**

Репозиторий — это централизованное хранилище пакетов ПО и их метаданных. Он позволяет системе автоматически находить, устанавливать и обновлять программы. Например, команда `apt update` обновляет список доступных пакетов из репозитория.

2. **В чем отличие онлайн репозитория от локального?**

Онлайн-репозиторий доступен через интернет (например, официальные серверы Ubuntu), а локальный хранится на жестком диске или в локальной сети. Локальный репозиторий используется для офлайн-установки, например, через файл `local.deb`.

3. **Что такое АРТ?**

АРТ (Advanced Package Tool) — система управления пакетами в Debian/Ubuntu, которая автоматически разрешает зависимости. Пример: `apt install firefox` установит браузер и все необходимые библиотеки.

4. **Для чего в лабораторной создавали файл .sig?**

Файл `.sig` содержит цифровую подпись для проверки целостности и подлинности данных. В лабораторной он использовался для безопасной передачи запроса обновлений через `apt-offline`.

5. **Что значит кроссплатформенный?**

Кроссплатформенное ПО работает на разных операционных системах. Например, `apt-offline` можно запустить в Windows через Python:

```
bash
```

Copy

```
python -m apt_offline get update.sig
```

6. **Как можно выполнить apt-offline в Windows?**

Установите Python для Windows и запустите `apt-offline` как модуль. Пример команды:

```
bash
```

Copy

```
python -m apt_offline install update.zip
```

7. **Расшифруйте команду: `sudo apt-offline get apt-offline.sig --threads 5 --bundle apt-offline-bundle.zip`**

Команда генерирует файл запроса (`apt-offline.sig`) с использованием 5 потоков для ускорения и упаковывает данные в ZIP-архив. Это нужно для переноса запроса на машину с интернетом.

8. **Как включить режим полного доступа к папке и файлу?**

Используйте команду `chmod 777` для выдачи прав чтения, записи и выполнения всем пользователям. Пример:

```
bash
```

Copy

```
sudo chmod 777 /home/user/docs/
```

Внимание: это снижает безопасность системы!

9. **Что содержит в себе файл `sources.list`?**

В `sources.list` хранятся адреса онлайн-репозитория. Пример строки:

```
plaintext
```

Copy

```
deb http://archive.ubuntu.com/ubuntu jammy main
```

Здесь указаны версия ОС (jammy) и раздел репозитория (main).

10. **Для чего необходимо было создавать файл `local.list` в директории `/sources.list.d/`?**

Файл `local.list` добавляет локальный репозиторий (например, для офлайн-установки). Система автоматически читает все файлы из `/etc/apt/sources.list.d/`, что удобнее прямого редактирования `sources.list`.

11. **Что такое зависимость пакетов и какую информацию она нам дает?**

Зависимости — это другие пакеты, необходимые для работы программы (например, библиотеки). Они указываются в метаданных пакета и позволяют `apt` автоматически устанавливать всё требуемое. Пример: пакет `gimp` зависит от `libgtk-3-0`.