

Лабораторная работа №5

Отчёт по лабораторной работе №5

Явкина Анастасия Юрьевна

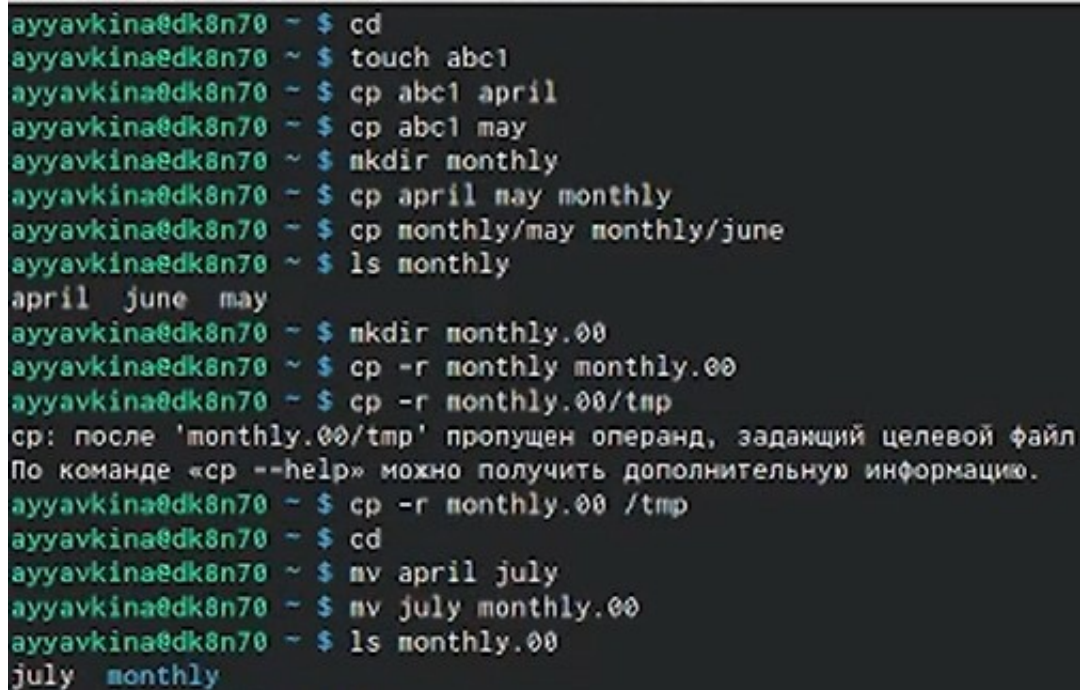
Содержание

Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

Выполнение лабораторной работы

1. Выполним примеры, приведённые в первой части описания лабораторной работы.



```
ayyavkina@dk8n70 ~ $ cd
ayyavkina@dk8n70 ~ $ touch abc1
ayyavkina@dk8n70 ~ $ cp abc1 april
ayyavkina@dk8n70 ~ $ cp abc1 may
ayyavkina@dk8n70 ~ $ mkdir monthly
ayyavkina@dk8n70 ~ $ cp april may monthly
ayyavkina@dk8n70 ~ $ cp monthly/may monthly/june
ayyavkina@dk8n70 ~ $ ls monthly
april  june  may
ayyavkina@dk8n70 ~ $ mkdir monthly.00
ayyavkina@dk8n70 ~ $ cp -r monthly monthly.00
ayyavkina@dk8n70 ~ $ cp -r monthly.00/tmp
cp: после 'monthly.00/tmp' пропущен операнд, задавший целевой файл
По команде «cp --help» можно получить дополнительную информацию.
ayyavkina@dk8n70 ~ $ cp -r monthly.00 /tmp
ayyavkina@dk8n70 ~ $ cd
ayyavkina@dk8n70 ~ $ mv april july
ayyavkina@dk8n70 ~ $ mv july monthly.00
ayyavkina@dk8n70 ~ $ ls monthly.00
july  monthly
```

Рис.1

```

ayyavkina@dk8n70 ~ $ ls
0.png      1.7.jpg      bin          monthly.00
1.10.jpg   1.8.jpg      GNUstep     Nastya
1.11.jpg   1.9.jpg      kl          Nastya.cpp
1.1.jpg    1.png       lab         Nastya.spp
1.2.jpg    '2022-04-22 13-59-03.mkv' lab1.cpp    op.pp
1.3.jpg    '2022-04-22 13-59-18.mkv' lab.cpp     public
1.4.jpg    2.png       lk          public_html
1.5.jpg    abc1        may         Screenshot_20210908_172111.png
1.6.jpg    Architecture_PC monthly      Screenshot_20210908_172534.png

```

Puc.2

```

ayyavkina@dk8n70 ~ $ mv monthly.00 monthly.01
ayyavkina@dk8n70 ~ $ mkdir reports
ayyavkina@dk8n70 ~ $ mv monthly.01 reports
ayyavkina@dk8n70 ~ $ mv reports/monthly.01 reports/monthly
ayyavkina@dk8n70 ~ $ cd
ayyavkina@dk8n70 ~ $ touch may
ayyavkina@dk8n70 ~ $ ls -l may
-rw-r--r-- 1 ayyavkina studsci 0 мая 6 16:42 may
ayyavkina@dk8n70 ~ $ chmod u+x may
ayyavkina@dk8n70 ~ $ ls -l may
-rwxr--r-- 1 ayyavkina studsci 0 мая 6 16:42 may
ayyavkina@dk8n70 ~ $ chmod u-x may
ayyavkina@dk8n70 ~ $ ls -l may
-rw-r--r-- 1 ayyavkina studsci 0 мая 6 16:42 may
ayyavkina@dk8n70 ~ $ cd
ayyavkina@dk8n70 ~ $ mkdir monthly

```

Puc.3

```

ayyavkina@dk8n70 ~ $ chmod g-r monthly
ayyavkina@dk8n70 ~ $ chmod o-r monthly
ayyavkina@dk8n70 ~ $ ls -l monthly
итого 0
-rw-r--r-- 1 ayyavkina studsci 0 мая 6 16:28 april
-rw-r--r-- 1 ayyavkina studsci 0 мая 6 16:29 june
-rw-r--r-- 1 ayyavkina studsci 0 мая 6 16:28 may

```

Puc.4

```

ayyavkina@dk8n70 ~ $ cd
ayyavkina@dk8n70 ~ $ touch abc1
ayyavkina@dk8n70 ~ $ chmod g+w abc1
ayyavkina@dk8n70 ~ $ ls -l abc1
-rw-rw-r-- 1 ayyavkina studsci 0 мая 6 17:01 abc1
ayyavkina@dk8n70 ~ $ mount
proc on /proc type proc (rw,relatime)
none on /run type tmpfs (rw,nosuid,nodev,relatime,me
udev on /dev type devtmpfs (rw,nosuid,relatime,size
devpts on /dev/pts type devpts (rw,relatime,gid=5,me
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,noexec
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,rel
/dev/sda8 on / type ext4 (rw,relatime)

```

Рис.5

2. Выполняем следующие действия:

- 1) Копируем файл /usr/include/sys/io.h в домашний каталог и называем его equipment.

```
ayyavkina@dk8n70 ~ $ cp /usr/include/sys/io.h equipment
```

Рис.9

- 2) В домашнем каталоге создаём директорию ~/ski.plases.

```
ayyavkina@dk8n70 ~ $ mkdir ski.plases
```

Рис.10

- 3) Переместим файл equipment в каталог ~/ski.plases.

```
ayyavkina@dk8n70 ~ $ mv equipment ski.plases
```

Рис.11

- 4) Переименуем файл ~/ski.plases/equipment в ~/ski.plases/equiplist. (Рис.20)

```
ayyavkina@dk8n70 ~ $ mv ski.plases/equipment ski.plases/equiplist
```

Рис.12

- 5) Создаём в домашнем каталоге файл abc1 и копируем его в каталог ~/ski.plases, называем его equiplist2. (

```
ayyavkina@dk8n70 ~ $ cp abc1 ski.plases/equiplist2
```

Рис.13

- 6) Создаём каталог с именем equipment в каталоге ~/ski.plases.


```
ayyavkina@dk8n70 ~ $ cd ski.plases
ayyavkina@dk8n70 ~/ski.plases $ mkdir equipment
```

Рис.14

- 7) Переместим файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.

```
ayyavkina@dk8n70 ~ $ mv ski.plases/equiplist ski.plases/equipment
ayyavkina@dk8n70 ~ $ mv ski.plases/equiplist2 ski.plases/equipment
```

Рис.15

- 8) Создадим и переместим каталог ~/newdir в каталог ~/ski.plases и назовём его plans.

```
ayyavkina@dk8n70 ~ $ mkdir newdir
ayyavkina@dk8n70 ~ $ mv newdir ski.plases
ayyavkina@dk8n70 ~ $ mv ski.plases/newdir plans
```

Рис.16

3. Определим опции команды chmod, необходимые для того, чтобы присвоить файлам и каталогам определённые права доступа.
- 1) для каталога australia

```
ayyavkina@dk8n70 ~ $ mkdir australia
ayyavkina@dk8n70 ~ $ mkdir play
ayyavkina@dk8n70 ~ $ chmod go-x australia
ayyavkina@dk8n70 ~ $ chmod g-r play
ayyavkina@dk8n70 ~ $ chmod o-r play
```

Рис.17

```
drwxr--r-- 2 ayyavkina studsci 2048 мая 6 17:44 australia
```

Рис.18

- 2) для каталога play

```
drwx--x--x 2 ayyavkina studsci 2048 мая 6 17:44 play
```

Рис.19

- 3) для файла my_os

```
ayyavkina@dk8n70 ~ $ touch my_os
ayyavkina@dk8n70 ~ $ touch feathers
```

Рис.20

```

ayyavkina@dk8n70 ~ $ chmod u-w my_os
ayyavkina@dk8n70 ~ $ chmod u+x my_os
ayyavkina@dk8n70 ~ $ ls -l my_os
-r-xr--r-- 1 ayyavkina studsci 0 мая 6 17:25 my_os

```

Рис.21

4) для файла feathers

```

ayyavkina@dk8n70 ~ $ chmod g+w feathers
ayyavkina@dk8n70 ~ $ ls -l feathers
-rw-rw-r-- 1 ayyavkina studsci 0 мая 6 17:25 feathers

```

Рис.22

1) Посмотрим содержимое файла /etc/fstab.

```

ayyavkina@dk8n70 ~ $ cat /etc/fstab
# /etc/fstab: static file system information.
# See the manpage fstab(5) for more information.

# <fs>          <mountpoint>    <type>          <options>
# NOTE: If your BOOT partition is ReiserFS, add the notail
LABEL=boot       /boot            ext4             re
LABEL=EFI         /boot/efi        vfat             re
LABEL=root        /                ext4             re
LABEL=swap        none             swap             sw
LABEL=afscache    /var/cache/openafs  ext4             re

#
proc             /proc            proc             de
shm              /dev/shm         tmpfs            no

```

Рис.6

2) Скопируем файл ~/feathers в файл ~/file.old.

```

ayyavkina@dk8n70 ~ $ cp feathers file.old

```

Рис.24

3) Переместим файл ~/file.old в каталог ~/play.

```

ayyavkina@dk8n70 ~ $ mv file.old play

```

Рис.25

4) Скопируем каталог ~/play в каталог ~/fun.

```
ayyavkina@dk8n70 ~ $ cp -r play fun
```

Рис.26

- 5) Переместим каталог ~/fun в каталог ~/play и назовём его games.

```
ayyavkina@dk8n70 ~ $ mv fun play
ayyavkina@dk8n70 ~ $ mv play/fun games
```

Рис.27

- 6) Лишим владельца файла ~/feathers права на чтение. При попытке просмотреть файл консоль выдаёт ошибку.

```
ayyavkina@dk8n70 ~ $ chmod u-r feathers
ayyavkina@dk8n70 ~ $ cat feathers
cat: feathers: Отказано в доступе
ayyavkina@dk8n70 ~ $ cp feathers at
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
ayyavkina@dk8n70 ~ $ chmod u+r feathers
ayyavkina@dk8n70 ~ $ cat feathers
```

Рис.28

- 7) Дадим владельцу файла ~/feathers право на чтение.

```
ayyavkina@dk8n70 ~ $ chmod u-r feathers
ayyavkina@dk8n70 ~ $ cat feathers
cat: feathers: Отказано в доступе
ayyavkina@dk8n70 ~ $ cp feathers at
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
ayyavkina@dk8n70 ~ $ chmod u+r feathers
ayyavkina@dk8n70 ~ $ cat feathers
```

Рис.28

- 8) Лишим владельца каталога ~/play права на выполнение. Попробуем перейти в каталог, не получается.

```
ayyavkina@dk8n70 ~ $ chmod u-x play
ayyavkina@dk8n70 ~ $ cd play
ayyavkina@dk8n70 ~/play $ cd
ayyavkina@dk8n70 ~ $ chmod u+x play
```

Рис.29

- 9) Вернём в каталог ~/play право на выполнение, мы смогли перейти в каталог.

```

ayyavkina@dk8n70 ~ $ chmod u-x play
ayyavkina@dk8n70 ~ $ cd play
ayyavkina@dk8n70 ~/play $ cd
ayyavkina@dk8n70 ~ $ chmod u+x play

```

Рис.29

- Используем команду `man` для просмотра описания следующих команд: `mount`, `fsck`, `mkfs`, `kill`.

```

ayyavkina@dk8n70 ~ $ man mount
ayyavkina@dk8n70 ~ $ man fsck
ayyavkina@dk8n70 ~ $ man mkfs
ayyavkina@dk8n70 ~ $ man kill

```

30

С помощью команды `mount` можно подключить сетевой диск, раздел жесткого диска или USB-накопитель. Команда `fsck` позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Команда `mkfs` делает файловые системы. Команда `kill` посылает сигнал процессу. Обычно используется для «убийства» процесса (прерывание процесса).

Выводы

- Я ознакомилась с файловой системой Linux, её структурой, именами и содержанием.
- Приобрела практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы каталогов.

Контрольные вопросы

- Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «`df-Th`». На моем компьютере есть следующие файловые системы: `dev tmpfs`, `tmpfs`, `ext4`, `iso9660`. `dev tmpfs` позволяет ядру создать экземпляр `tmpfs` с именем `devtmpfs` при инициализации ядра, прежде чем регистрируется какое-либо устройство с драйверами. Каждое устройство с майором / минором будет предоставлять узел устройства в `devtmpfs`. `devtmpfs` монтируется на `/dev` и содержит специальные файлы устройств для всех устройств. `tmpfs` – временное файловое хранилище во многих Unix-подобных ОС. Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо ПЗУ. Подобная конструкция является RAM диском. Данная файловая система также

предназначенная для быстрого и ненадёжного хранения временных данных. Хорошо подходит для /tmp и массовой сборки пакетов/образов. Предполагает наличие достаточного объёма виртуальной памяти. Файловая система tmpfs предназначена для того, чтобы использовать часть физической памяти сервера как обычный дисковый раздел, в котором можно сохранять данные (чтение и запись). Поскольку данные размещены в памяти, то чтение или запись происходят во много раз быстрее, чем с обычного HDD диска. ext4 – имеет обратную совместимость с предыдущими версиями ФС. Эта версия была выпущена в 2008 году. Является первой ФС из «семейства» Ext, использующая механизм «extentfile system», который позволяет добиться меньшей фрагментации файлов и увеличить общую производительность файловой системы. Кроме того, в Ext4 реализован механизм отложенной записи (delayed allocation – delalloc), который так же уменьшает фрагментацию диска и снижает нагрузку на CPU. С другой стороны, хотя механизм отложенной записи и используется во многих ФС, но в силу сложности своей реализации он повышает вероятность утери данных. Характеристики: максимальный размер файла: 16 TB; максимальный размер раздела: 16TB; максимальный размер имени файла: 255 символов. Рекомендации по использованию: наилучший выбор для SSD; наилучшая производительность по сравнению с предыдущими Ext-системами; она так же отлично подходит в качестве файловой системы для серверов баз данных, хотя сама система и моложе Ext3. ISO 9660 – стандарт, выпущенный Международной организацией по стандартизации, описывающий файловую систему для дисков CD-ROM. Также известен как CDFS (Compact Disc File System). Целью стандарта является обеспечить совместимость носителей под разными операционными системами, такими, как Unix, Mac OS, Windows.

2. Файловая система Linux/UNIX физически представляет собой пространство раздела диска разбитое на блоки фиксированного размера, кратные размеру сектора – 1024, 2048, 4096 или 8120 байт. Размер блока указывается при создании файловой системы. В файловой структуре Linux имеется один корневой раздел – / (он же root, корень). Все разделы жесткого диска (если их несколько) представляют собой структуру подкаталогов, “примонтированных” к определенным каталогам.

“/” – корень. Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. “/BIN” – бинарные файлы пользователя. Этот каталог содержит исполняемые файлы. “/SBIN” – системные исполняемые файлы. Так же как и “/bin”, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог /usr. “/ETC” – конфигурационные файлы. В этой папке содержатся конфигурационные файлы всех

программ, установленных в системе. “/DEV” – файлы устройств. В Linux все, в том числе внешние устройства являются файлами. “/PROC” – информация о процессах. По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его Pid, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. “/VAR” – переменные файлы. Он содержит файлы, которые часто изменяются. “/TMP” – временные файлы. В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. “/USR” – программы пользователя. Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыку и документацию. “/HOME” – домашняя папка. В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т.д. “/BOOT” – файлы загрузчика. Содержит все файлы, связанные с загрузчиком системы. Это ядро vmlinuz, образ initrd, а также файлы загрузчика, находящиеся в каталоге /boot/grub. “/LIB” – системные библиотеки. Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах /bin и /sbin. “/OPT” – дополнительные программы. В эту папку устанавливаются программы, игры или драйвера. Это программы созданные в виде отдельных исполняемых файлов самими производителями. “/MNT” – монтирование. В этот каталог системные администраторы могут монтировать внешние или дополнительные файловые системы. “/MEDIA” – съемные носители. В этот каталог система монтирует все подключаемые внешние накопители –USB флешки, оптические диски и другие носители информации. “/SRV” – сервер. В этом каталоге содержатся файлы серверов и сервисов. “/RUN” - процессы. Каталог, содержащий PID файлы процессов, похожий на “/var/run”, но в отличие от него, он размещен в TMPFS, а поэтому после перезагрузки все файлы теряются.

3. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой mount.
4. Чтобы устранить повреждения файловой системы необходимо использовать команду fsck.
5. Файловую систему можно создать, используя команду mkfs. Ее краткое описание дано в пункте 5 в ходе выполнения заданий лабораторной работы.
6. Для просмотра текстовых файлов существуют следующие команды:

cat. Задача команды cat очень проста –она читает данные из файла или стандартного ввода и выводит их на экран. Синтаксис утилиты: cat [опции] файл1 файл2 ...Основные опции: -b-нумеровать только непустые строки, -E-

показывать символ \$ в конце каждой строки, -n-нумеровать все строки, -s-удалять пустые повторяющиеся строки, -T-отображать табуляции в виде ^I, -h-отобразить справку, -v-версия утилиты nl. Команда nl действует аналогично команде cat, но выводит еще и номера строк в столбце слева. less. Существенно более развитая команда для пролистывания текста. При чтении данных со стандартного ввода она создает буфер, который позволяет листать текст как вперед, так и назад, а также искать как по направлению к концу, так и по направлению к началу текста. Синтаксис аналогичный синтаксису команды cat. Некоторые опции: -g -при поиске подсвечивать только текущее найденное слово (по умолчанию подсвечиваются все вхождения), -N -показывать номера строк head. Команда head выводит начальные строки (по умолчанию - 10) из одного или нескольких документов. Также она может показывать данные, которые передает на вывод другая утилита. Синтаксис аналогичный синтаксису команды cat. Основные опции: -c (-bytes) -позволяет задавать количество текста не в строках, а в байтах -n (-lines) -показывает заданное количество строк вместо 10, которые выводятся по умолчанию -q (-quiet, -silent) -выводит только текст, недобавляя к нему название файла -v (-verbose) -перед текстом выводит название файла -z (-zero-terminated) -символы перехода на новую строку заменяет символами завершения строк tail. Эта команда позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме. Синтаксис аналогичный синтаксису команды cat. Основные опции: -c -выводить указанное количество байт с конца файла, -f -обновлять информацию по мере появления новых строк в файле, -n -выводить указанное количество строк из конца файла, -pid -используется с опцией -f, позволяет завершить работу утилиты, когда завершится указанный процесс, -q -не выводить имена файлов -retry- повторять попытки открыть файл, если он недоступен, -v -выводить подробную информацию о файле.

7. Утилита cp позволяет полностью копировать файлы и директории. Синтаксис: cp [опции] файл-источник файл-приемник. После выполнения команды файл-источник будет полностью перенесен в файл-приемник. Если в конце указан слэш, файл будет записан в заданную директорию с оригинальным именем. Основные опции: -attributes-only -не копировать содержимое файла, а только флаги доступа и владельца -f, -force -перезаписывать существующие файлы -i, -interactive -спрашивать, нужно ли перезаписывать существующие файлы, -L -копировать не символические ссылки, а то, на что они указывают, -n -не перезаписывать существующие файлы, -P -не следовать символическим ссылкам, -r -копировать папку Linux рекурсивно, -s -не выполнять копирование файлов в Linux, а создавать символические ссылки, -u -скопировать файл, только если он был изменён, -x -не выходить за пределы этой файловой системы, -p -сохранять владельца, временные метки и флаги доступа при копировании, -t -считать файл-приемник директорией и копировать файл-источник в эту директорию.

8. Команда `mv` используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий. Синтаксис: `mv [-опции] старый_файл новый_файл`. Основные опции: `-help` – выводит на экран официальную документацию об утилите, `-version` – отображает версию, `mv -b` – создает копию файлов, которые были перемещены или перезаписаны, `-f` – при активации не будет спрашивать разрешение у владельца файла, если речь идет о перемещении или переименовании файла, `-i` – наоборот, будет спрашивать разрешение у владельца, `-n` – отключает перезапись уже существующих объектов `-strip-trailing-slashes` — удаляет завершающий символ `/` у файла при его наличии, `-t [директория]` — перемещает все файлы в указанную директорию, `-u` – осуществляет перемещение только в том случае, если исходный файл новее объекта назначения, `-v` – отображает сведения о каждом элементе во время обработки команды. Команда `rename` также предназначена, чтобы переименовать файл. Синтаксис: `rename [опции] старое_имя новое_имя файлы`. Основные опции: `-v` – вывести список обработанных файлов, `-n` – тестовый режим, на самом деле никакие действия выполнены не будут, `-f` – принудительно перезаписывать существующие файлы.
9. Права доступа – совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Синтаксис команды: `chmod режим имя_файла`. Режим имеет следующие компоненты структуры и способ записи: `=` установить право, `-` лишить права, `+` дать право, `r` чтение, `w` запись, `x` выполнение, `u` (user) владелец файла, `g` (group) группа, к которой принадлежит владелец файлов, `o` (others) все остальные.