

Лабораторная работа №5

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Нилова Кристина

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Выводы	15

Список иллюстраций

3.1	выполнение команд	8
3.2	выполнение команд	8
3.3	выполнение команд	9
3.4	выполнение команд	9
3.5	результат	10
3.6	результат	10
3.7	результат	10
3.8	результат	10
3.9	выполнение команд	11
3.10	выполнение команд	11
3.11	man	12
3.12	man mount	12
3.13	man fsck	13
3.14	man mkfs	13
3.15	man kill	14

Список таблиц

1 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.

2 Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.

2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

2.1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него. 2.2. В домашнем каталоге создайте директорию `~/ski.places`. 2.3. Переместите файл `equipment` в каталог `~/ski.places`. 2.4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`. 2.5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`. 2.6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`. 2.7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`. 2.8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.

3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет: 3.1. `drwxr-r- ... australia` 3.2. `drwx-x-x ... play` 3.3. `-r-xr-r- ... my_os` 3.4. `-rw-rw-r- ... feathers` При необходимости создайте нужные файлы.

4. Прделаем приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

4.1. Просмотрите содержимое файла `/etc/passwd`. 4.2. Скопируйте файл `~/feathers` в файл `~/file.old`. 4.3. Переместите файл `~/file.old` в каталог `~/play`. 4.4. Скопируйте каталог `~/play` в каталог `~/fun`. 4.5. Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`. 4.6. Лишите владельца файла `~/feathers` права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`? 4.8. Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`? 4.9. Дайте владельцу файла `~/feathers` право на чтение. 4.10. Лишите владельца каталога `~/play` права на выполнение. 4.11. Перейдите в каталог `~/play`. Что произошло? 4.12. Дайте владельцу каталога `~/play` право на выполнение. 5. Прочитайте `man` по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

3 Выполнение лабораторной работы

1. Выполняем все примеры, приведённые в первой части описания лабораторной работы.

```
kanilova@dk2n25 ~ $ mkdir feathers
kanilova@dk2n25 ~ $ cd feathers
kanilova@dk2n25 ~/feathers $ touch file.old
kanilova@dk2n25 ~/feathers $ cd
kanilova@dk2n25 ~ $ cp -r ~/feathers ~/file.old
kanilova@dk2n25 ~ $ mv ~/file.old ~/play
kanilova@dk2n25 ~ $ cp -r ~/play ~/fun
kanilova@dk2n25 ~ $ mv ~/fun/play ~/fun/games
kanilova@dk2n25 ~ $ mv ~/fun/play ~/fun/games
mv: не удалось выполнить stat для '/afs/.dk.sci.pfu.edu.ru/home/k/a/kanilova/fun/play': Нет такого файла и
ли каталога
kanilova@dk2n25 ~ $ ls
arch-pc  dir1  feathers  parentdir  parentdir2  public_html  Доклад.md  Общедоступные
bin      dir2  GNUstep  parentdir3  parentdir3  tmp          Загрузки   'Рабочий стол'
chimp   dir3  hosts    parentdir  play        work         Изображения 'Снимки экрана'
chimp.pub Documents newdir1  parentdir1  public      Видео       Музыка     Шаблоны
kanilova@dk2n25 ~ $ cd play
kanilova@dk2n25 ~/play $ mv fun games
kanilova@dk2n25 ~/play $ ls
file.old  games
kanilova@dk2n25 ~/play $
```

Рис. 3.1: выполнение команд

```
kanilova@dk2n25 ~/play $ cd
kanilova@dk2n25 ~ $ cp /usr/include/sys/io.h equipment
kanilova@dk2n25 ~ $ ls
arch-pc  dir2  GNUstep  parentdir  public  Доклад.md  'Рабочий стол'
bin      dir3  hosts    parentdir1  public_html  Загрузки   'Снимки экрана'
chimp   Documents newdir1  parentdir2  tmp        Изображения Шаблоны
chimp.pub equipment parentdir parentdir3  work       Музыка
dir1    feathers parentdir3 play        Видео      Общедоступные
kanilova@dk2n25 ~ $ mkdir ~/ski.plases
kanilova@dk2n25 ~ $ ls
arch-pc  dir2  GNUstep  parentdir  public  Видео  Общедоступные
bin      dir3  hosts    parentdir1  public_html  Доклад.md  'Рабочий стол'
chimp   Documents newdir1  parentdir2  ski.plases  Загрузки   'Снимки экрана'
chimp.pub equipment parentdir parentdir3  tmp        Изображения Шаблоны
dir1    feathers parentdir3 play        work       Музыка
kanilova@dk2n25 ~ $ mv equipment ski.plases
kanilova@dk2n25 ~ $ mv ~/ski.plases/equipment ~/ski.plases/equiplist
kanilova@dk2n25 ~ $ ls ski.plases
equiplist
kanilova@dk2n25 ~ $
```

Рис. 3.2: выполнение команд

2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

2.1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него. 2.2. В домашнем каталоге создайте директорию `~/ski.plases`. 2.3. Переместите файл `equipment` в каталог `~/ski.plases`. 2.4. Переименуйте файл `~/ski.plases/equipment` в `~/ski.plases/equiplist`.

```
equiplist
kanilova@dk2n25 ~ $ touch abc1
kanilova@dk2n25 ~ $ cp -r ~/ski.plases equiplist2
kanilova@dk2n25 ~ $ cd ~/ski.plases
kanilova@dk2n25 ~/ski.plases $ ls
equiplist
kanilova@dk2n25 ~/ski.plases $ cp ~/abc1 equiplist2
kanilova@dk2n25 ~/ski.plases $ ls
equiplist equiplist2
kanilova@dk2n25 ~/ski.plases $ mkdir equipment
kanilova@dk2n25 ~/ski.plases $ ls
equiplist equiplist2 equipment
kanilova@dk2n25 ~/ski.plases $ mv equiplist equiplist2 equipment
kanilova@dk2n25 ~/ski.plases $ ls equipment
equiplist equiplist2
kanilova@dk2n25 ~/ski.plases $ mkdir ~/newdir
kanilova@dk2n25 ~/ski.plases $ mv ~/newdir plans
kanilova@dk2n25 ~/ski.plases $ ls
equipment plans
kanilova@dk2n25 ~/ski.plases $
```

Рис. 3.3: выполнение команд

2.5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.plases`, назовите его `equiplist2`. 2.6. Создайте каталог с именем `equipment` в каталоге `~/ski.plases`. 2.7. Переместите файлы `~/ski.plases/equiplist` и `equiplist2` в каталог `~/ski.plases/equipment`. 2.8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.plases` и назовите его `plans`.

```
kanilova@dk2n25 ~ $ mkdir australia
kanilova@dk2n25 ~ $ chmod g-x australia
kanilova@dk2n25 ~ $ chmod o-x australia
kanilova@dk2n25 ~ $ ls -l australia
итого 0
kanilova@dk2n25 ~ $ ls -l
итого 86
-rw-r--r-- 1 kanilova studsci  0 мар 15 16:35 abc1
drwxr-xr-x 5 kanilova studsci 2048 фев 7 23:40 arch-pc
drwxr--r-- 2 kanilova studsci 2048 мар 15 16:39 australia
```

Рис. 3.4: выполнение команд

3. Определим опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в

начале таких прав нет:

3.1. drwxr-r- ... australia

```
drwxr-xr-x 2 kanilova studsci 2048 сен 7 2023
kanilova@dk2n25 ~ $ chmod o-r play
kanilova@dk2n25 ~ $ s -l
bash: s: команда не найдена
kanilova@dk2n25 ~ $ ls -l
итого 86
-rw-r--r-- 1 kanilova studsci 0 мар 15 16:25
```

Рис. 3.5: результат

3.2. drwx-x-x ... play

```
kanilova@dk2n25 ~ $ chmod u-w my_os
kanilova@dk2n25 ~ $ chod u+w my_os
bash: chod: команда не найдена
kanilova@dk2n25 ~ $ chmod u+w my_os
kanilova@dk2n25 ~ $ ls -l
итого 88
-rw-r--r-- 1 kanilova studsci 0 мар 15 16:35 abc1
drwxr-xr-x 5 kanilova studsci 2048 фев 7 23:40 arch-pc
drwxr--r-- 2 kanilova studsci 2048 мар 15 16:39 australia
```

Рис. 3.6: результат

3.3. -r-xr-r- ... my_os

```
drwxr-xr-x 2 kanilova studsci 2048 сен 7 2023 шаблоны
kanilova@dk2n25 ~ $ cat ~/feathers
cat: /afs/.dk.sci.pfu.edu.ru/home/k/a/kanilova/feathers: Это каталог
kanilova@dk2n25 ~ $
```

Рис. 3.7: результат

3.4. -rw-rw-r- ... feathers

```
kanilova@dk2n25 ~ $ cp feathers
cp: после 'feathers' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
```

Рис. 3.8: результат

4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

4.1. Просмотрите содержимое файла /etc/passwd. 4.2. Скопируйте файл ~/feathers в файл ~/file.old. 4.3. Переместите файл ~/file.old в каталог ~/play. 4.4. Скопируйте каталог ~/play в каталог ~/fun. 4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games.

```
kanilova@dk2n25 ~ $ chmod u-r feathers
kanilova@dk2n25 ~ $ chmod u+r feathers
chmod: невозможно получить доступ к 'feathers': Нет такого файла или каталога
kanilova@dk2n25 ~ $ chmod u+r feathers
kanilova@dk2n25 ~ $
```

Рис. 3.9: выполнение команд

4.6. Лишите владельца файла ~/feathers права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?

```
kanilova@dk2n25 ~ $ chmod u-x ~/play
kanilova@dk2n25 ~ $ cd play
kanilova@dk2n25 ~/play $ ls -l
итого 2
-rw-r--r-- 1 kanilova studsci  0 мар 15 16:27 file.old
drwxr-xr-x 2 kanilova studsci 2048 мар 15 16:28 games
kanilova@dk2n25 ~/play $
```

4.8. Что произойдёт,

если вы попытаетесь скопировать файл ~/feathers?

```
kanilova@dk2n25 ~/play $ cd
kanilova@dk2n25 ~ $ chmod u+x ~/play
kanilova@dk2n25 ~ $ ls -l
итого 88
-rw-r--r-- 1 kanilova studsci  0 мар 15 16:35 abc1
drwxr-xr-x 5 kanilova studsci 2048 фев  7 23:40 arch-pc
drwxr--r-- 2 kanilova studsci 2048 мар 15 16:39 australia
drwxr-xr-x 2 kanilova studsci 2048 мар  6 15:48 bin
-rw----- 1 kanilova studsci 2622 окт 27 15:22 chimp
-rw-r--r-- 1 kanilova studsci  583 окт 27 15:22 chimp.pub
drwxr-xr-x 2 kanilova studsci 2048 сен 14 2023 dir1
drwxr-xr-x 2 kanilova studsci 2048 сен 14 2023 dir2
drwxr-xr-x 2 kanilova studsci 2048 сен 14 2023 dir3
drwxr-xr-x 2 kanilova studsci 2048 окт  9 19:40 Documents
drwxr-xr-x 2 kanilova studsci 2048 мар 15 16:36 equiplist2
drwxr-xr-x 2 kanilova studsci 2048 мар 15 16:26 feathers
drwxr-xr-x 2 kanilova studsci 2048 мар  5 16:53 file.old
```

Рис. 3.10: выполнение команд

4.9. Дайте владельцу файла ~/feathers право на чтение.

4.10. Лишите владельца каталога ~/play права на выполнение.

4.11. Перейдите в каталог ~/play. Что произошло? ничего не произошло(

4.12. Дайте владельцу каталога ~/play право на выполнение.

!

5. Прочитаем man по командам mount, fsck, mkfs, kill и кратко их охарактери-
зуем, приведя примеры.

```
kanilova@dk2n25 ~ $ man mount
kanilova@dk2n25 ~ $ man fsck
kanilova@dk2n25 ~ $ man mkfs
kanilova@dk2n25 ~ $ man kill
kanilova@dk2n25 ~ $
```

Рис. 3.11: man

```
mount(8) System Administration mount(8)
NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount --make-[shared|slave|private|unbindable|rshared|rsave|rprivate|runbindable] mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted
    at /. These files can be spread out over several devices. The mount command serves to attach the
    filesystem found on some device to the big file tree. Conversely, the umount(8) command will
    detach it again. The filesystem is used to control how data is stored on the device or provided
    in a virtual way by network or other services.

    The standard form of the mount command is:

        mount -t type device dir

    This tells the kernel to attach the filesystem found on device (which is of type type) at the
    directory dir. The option -t type is optional. The mount command is usually able to detect a
    filesystem. The root permissions are necessary to mount a filesystem by default. See section
    "Non-superuser mounts" below for more details. The previous contents (if any) and owner and mode
    of dir become invisible, and as long as this filesystem remains mounted, the pathname dir refers
    Manual page mount(8) line 1 (press h for help or q to quit)
```

Рис. 3.12: man mount

```
FSCK(8)                                     System Administration                                     FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--] [fs-specific-options]

DESCRIPTION
    fsck is used to check and optionally repair one or more Linux filesystems. filesystem can be a device name (e.g., /dev/hdc1, /dev/sdb2), a mount point (e.g., /, /usr, /home), or a filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root). Normally, the fsck program will try to handle filesystems on different physical disk drives in parallel to reduce the total amount of time needed to check all of them.

    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to checking filesystems in /etc/fstab serially. This is equivalent to the -As options.

    The exit status returned by fsck is the sum of the following conditions:

    0
        No errors

    1
        Filesystem errors corrected

    2
        System should be rebooted

    4
        Filesystem errors left uncorrected

    8
        Operational error

Manual page fsck(8) line 1 (press h for help or q to quit)
```

Рис. 3.13: man fsck

```
MKFS(8)                                     System Administration                                     MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard disk partition. The device argument is either the device name (e.g., /dev/hda1, /dev/sdb2), or a regular file that shall contain the filesystem. The size argument is the number of blocks to be used for the filesystem.

    The exit status returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem builders (mkfs.fstype) available under Linux. The filesystem-specific builder is searched for via your PATH environment setting only. Please see the filesystem-specific builder manual pages for further details.

OPTIONS
    -t, --type type
        Specify the type of filesystem to be built. If not specified, the default filesystem type (currently ext2) is used.

    fs-options
        Filesystem-specific options to be passed to the real filesystem builder.

    -V, --verbose
        Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.

    -h, --help
        Display help text and exit.

Manual page mkfs(8) line 1 (press h for help or q to quit)
```

Рис. 3.14: man mkfs

```

kill(1)                                User Commands                                kill(1)
NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include HUP, INT, KILL, STOP, CONT, and 0. Alternate signals may be specified in three ways: -9, -SIGKILL or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
    --signal <signal>
        Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

    -q, --queue value
        Use sigqueue(3) rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t structure.

    -l, --list [signal]
        List signal names. This option has optional argument, which will convert signal number to signal name, or other way round.

    -L, --table

Manual page kill(1) line 1 (press h for help or q to quit)

```

Рис. 3.15: man kill

4 Выводы

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