

Отчёта по лабораторной работе №5.

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

Гандич Дарья Владимировна. НБИбд-02-22.

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	15

Список иллюстраций

2.1	Пример 1	6
2.2	Пример 2	6
2.3	Пример 3	7
2.4	Пример 4	7
2.5	Пункт 2	8
2.6	australa	8
2.7	play	9
2.8	feathers, my-os	9
2.9	Пункт 4	10
2.10	mount	11
2.11	fsck	12
2.12	mkfs	13
2.13	kill	14
2.14	man	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 Выполнение лабораторной работы

1. Выполняем все примеры из текста ЛР: 1) копируем файл ~/abc1 в файл april и в файл may, 2) копируем каталог monthly в каталог monthly.00, 3) переименование файлов в текущем каталоге, 4) использование режимов команды chmod.

```
dvgandich@dk4n64 ~ $ cd
dvgandich@dk4n64 ~ $ touch abc1
dvgandich@dk4n64 ~ $ cp abc1 apri
dvgandich@dk4n64 ~ $ cp abc1 may
dvgandich@dk4n64 ~ $ mkdir monthly
dvgandich@dk4n64 ~ $ cp april may monthly
cp: не удалось выполнить stat для 'april': Нет такого файла или каталога
dvgandich@dk4n64 ~ $ cp abc1 april
dvgandich@dk4n64 ~ $ cp april may monthly
dvgandich@dk4n64 ~ $ cp monthly/may monthly/june
dvgandich@dk4n64 ~ $ ls monthly
april  june  may
dvgandich@dk4n64 ~ $
```

Рис. 2.1: Пример 1

```
dvgandich@dk4n64 ~ $ mkdir monthly.00
dvgandich@dk4n64 ~ $ cp -r monthly monthly.00
dvgandich@dk4n64 ~ $ cp -r monthly.00 /tmp
dvgandich@dk4n64 ~ $
```

Рис. 2.2: Пример 2

```

dvgandich@dk4n64 ~ $ mv april july
dvgandich@dk4n64 ~ $ mv july monthly.00
dvgandich@dk4n64 ~ $ ls monthly.00
july  monthly
dvgandich@dk4n64 ~ $ mv monthly.00 monthly.01
dvgandich@dk4n64 ~ $ mkdir reports
dvgandich@dk4n64 ~ $ mv monthly.01 reports
dvgandich@dk4n64 ~ $ mv reports/monthly.01 reports/monthly
dvgandich@dk4n64 ~ $

```

Рис. 2.3: Пример 3

```

dvgandich@dk4n64 ~ $ touch may
dvgandich@dk4n64 ~ $ ls -l may
-rw-r--r-- 1 dvgandich studsci 0 map  9 14:35 may
dvgandich@dk4n64 ~ $ chmod u+x may
dvgandich@dk4n64 ~ $ ls -l may
-rwxr--r-- 1 dvgandich studsci 0 map  9 14:35 may
dvgandich@dk4n64 ~ $ chmod u-x may
dvgandich@dk4n64 ~ $ ls -l may
-rw-r--r-- 1 dvgandich studsci 0 map  9 14:35 may
dvgandich@dk4n64 ~ $ cd
dvgandich@dk4n64 ~ $ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
dvgandich@dk4n64 ~ $ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
dvgandich@dk4n64 ~ $ cd
dvgandich@dk4n64 ~ $ touch abc1
dvgandich@dk4n64 ~ $ chmod g+w abc1
dvgandich@dk4n64 ~ $

```

Рис. 2.4: Пример 4

2. Выполняем пункт 2 из текста ЛР, вспоминаем работу команд `cp` и `mv`

```

dvgandich@dk4n64 ~ $ cp /usr/include/sys/io.h /home/dvgandich
cp: невозможно создать обычный файл '/home/dvgandich': Отказано в доступе
dvgandich@dk4n64 ~ $ cp /usr/include/sys/io.h /home/dvgandich
cp: невозможно создать обычный файл '/home/dvgandich': Отказано в доступе
dvgandich@dk4n64 ~ $ cp /usr/include/sys/io.h /home
cp: невозможно создать обычный файл '/home/io.h': Отказано в доступе
dvgandich@dk4n64 ~ $ cp /usr/include/sys/io.h /home/dvgandich
cp: невозможно создать обычный файл '/home/dvgandich': Отказано в доступе
dvgandich@dk4n64 ~ $ mkdir ~/ski.plases
dvgandich@dk4n64 ~ $ mv io.h equipment
dvgandich@dk4n64 ~ $ mv equipment ~/ski.plases
dvgandich@dk4n64 ~ $ mv ~/ski.plases/equipment ~/ski.plases/equiplist
dvgandich@dk4n64 ~ $ mkdir abc1
mkdir: невозможно создать каталог «abc1»: Файл существует
dvgandich@dk4n64 ~ $ cp abc1 ~/ski.plases
dvgandich@dk4n64 ~ $ mv ~/ski.plases/abc1 ~/ski.plases/equiplist2
dvgandich@dk4n64 ~ $ cd ski.plases
dvgandich@dk4n64 ~/ski.plases $ mkdir equipment
dvgandich@dk4n64 ~/ski.plases $ cd
dvgandich@dk4n64 ~ $ mv ~/ski.plases/equiplist ~/ski.plases/equiplist2 ~/ski.plases/equipment
dvgandich@dk4n64 ~ $ mv newdir ~/ski.plases
mv: не удалось выполнить stat для 'newdir': Нет такого файла или каталога
dvgandich@dk4n64 ~ $ mkdir newdir
dvgandich@dk4n64 ~ $ mv newdir ~/ski.plases
dvgandich@dk4n64 ~ $ mv ~/ski.plases/newdir ~/ski.plases/plans

```

Рис. 2.5: Пункт 2

3. Выполняем пункт 3 из текста ЛР, определяем опции команды `chmod`, необходимые для того, чтобы присвоить файлам `australia`, `play`, `my_os`, `feathers` выделенные права доступа. С файлами `my_os` и `feathers` достигнуть указанного права доступа не получилось.

```

dvgandich@dk4n64 ~ $ mkdir australia
dvgandich@dk4n64 ~ $ chmod g-x australia
dvgandich@dk4n64 ~ $ chmod o-x australia
dvgandich@dk4n64 ~ $ chmod g-w australia
dvgandich@dk4n64 ~ $ ls -l
итого 4209
-rw-rw-r-- 1 dvgandich studsci      0 мар  9 14:37 abc1
drwxr--r-- 2 dvgandich studsci  2048 мар  9 15:06 australia

```

Рис. 2.6: australa


```

dvgandich@dk4n64 ~ $ mkdir play
dvgandich@dk4n64 ~ $ chmod go-r,g-w play
dvgandich@dk4n64 ~ $ ls -l
итого 4211
-rw-rw-r-- 1 dvgandich studsci      0 мар  9 14:37 abc1
drwxr--r-- 2 dvgandich studsci    2048 мар  9 15:06 australia
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 bin
drwxr-xr-x 2 dvgandich studsci    2048 фев  4 20:29 CC+
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 hugo_extended_0.110.0_Linux-64bit
-rwxr-xr-x 1 dvgandich studsci   16224 сен 28 11:28 kol
-rw-r--r-- 1 dvgandich studsci     146 сен 28 11:27 kol.cpp
-rw-r--r-- 1 dvgandich studsci      0 мар  9 14:35 may
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 14:25 monthly
drwx--x--x 2 dvgandich studsci    2048 мар  9 15:08 play

```

Рис. 2.7: play

```

dvgandich@dk4n64 ~ $ touch feathers
dvgandich@dk4n64 ~ $ ls -l
итого 4211
-rw-rw-r-- 1 dvgandich studsci      0 мар  9 14:37 abc1
drwxr--r-- 2 dvgandich studsci    2048 мар  9 15:06 australia
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 bin
drwxr-xr-x 2 dvgandich studsci    2048 фев  4 20:29 CC+
-rw-r--r-- 1 dvgandich studsci      0 мар  9 15:15 feathers
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 hugo_extended_0.110.0_Linux-64bit
-rwxr-xr-x 1 dvgandich studsci   16224 сен 28 11:28 kol
-rw-r--r-- 1 dvgandich studsci     146 сен 28 11:27 kol.cpp
-rw-r--r-- 1 dvgandich studsci      0 мар  9 14:35 may
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 14:25 monthly
drwx--x--x 2 dvgandich studsci    2048 мар  9 15:08 play
-rw-r--r-- 1 dvgandich studsci    4995 фев 22 14:54 presentation.md
drwxr-xr-x 3 dvgandich root      2048 фев  4 20:29 public
lrwxr-xr-x 1 dvgandich root       18 мар  2 22:29 public_html -> public/public_html
-rw-r--r-- 1 dvgandich studsci 4251441 мар  9 14:44 report.pdf
drwxr-xr-x 3 dvgandich studsci    2048 мар  9 14:30 reports
drwxr-xr-x 4 dvgandich studsci    2048 мар  9 15:02 ski_places
drwxr-xr-x 2 dvgandich studsci    2048 сен 28 10:55 tmp
drwxr-xr-x 5 dvgandich studsci    2048 фев 22 16:18 work
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Видео
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Документы
drwxr-xr-x 3 dvgandich studsci    2048 ноя 16 11:39 Изображения
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Музыка
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Общедоступные
drwxr-xr-x 2 dvgandich studsci    2048 ноя  9 14:31 'Рабочий стол'
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Шаблоны
dvgandich@dk4n64 ~ $ mkdir my_os
dvgandich@dk4n64 ~ $ ls -l
итого 4213
-rw-rw-r-- 1 dvgandich studsci      0 мар  9 14:37 abc1
drwxr--r-- 2 dvgandich studsci    2048 мар  9 15:06 australia
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 bin
drwxr-xr-x 2 dvgandich studsci    2048 фев  4 20:29 CC+
-rw-r--r-- 1 dvgandich studsci      0 мар  9 15:15 feathers
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 hugo_extended_0.110.0_Linux-64bit
-rwxr-xr-x 1 dvgandich studsci   16224 сен 28 11:28 kol
-rw-r--r-- 1 dvgandich studsci     146 сен 28 11:27 kol.cpp
-rw-r--r-- 1 dvgandich studsci      0 мар  9 14:35 may
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 14:25 monthly
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 15:23 my_os

```

Рис. 2.8: feathers, my-os

4. Выполняем пункт 4 из текста ЛР, меняем права доступа к файлам play и feathers

```
dvgandich@dk4n64 ~ $ cp ~/feathers ~/file.old
dvgandich@dk4n64 ~ $ mv ~/file.old ~/play
dvgandich@dk4n64 ~ $ cp -r ~/play ~/fun
dvgandich@dk4n64 ~ $ mv ~/fun ~/play
dvgandich@dk4n64 ~ $ cd ~/play
dvgandich@dk4n64 ~/play $ mv fun games
dvgandich@dk4n64 ~/play $ cd
dvgandich@dk4n64 ~ $ chmod u-r feathers
dvgandich@dk4n64 ~ $ chmod u-x play
dvgandich@dk4n64 ~ $ cd ~/play
dvgandich@dk4n64 ~/play $ cd
dvgandich@dk4n64 ~ $ chmod u-x play
dvgandich@dk4n64 ~ $ cd ~/play
dvgandich@dk4n64 ~/play $ cd
dvgandich@dk4n64 ~ $ chmod u+x play
dvgandich@dk4n64 ~ $ ls -l
итого 4213
-rw-rw-r-- 1 dvgandich studsci      0 мар  9 14:37 abc1
drwxr--r-- 2 dvgandich studsci    2048 мар  9 15:06 australia
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 bin
drwxr-xr-x 2 dvgandich studsci    2048 фев  4 20:29 CC+
--w-r--r-- 1 dvgandich studsci      0 мар  9 15:27 feathers
drwxr-xr-x 2 dvgandich studsci    2048 фев 22 15:50 hugo_extended_0.110.0_Linux-64bit
-rwxr-xr-x 1 dvgandich studsci   16224 сен 28 11:28 kol
-rw-r--r-- 1 dvgandich studsci     146 сен 28 11:27 kol.cpp
-rw-r--r-- 1 dvgandich studsci      0 мар  9 14:35 may
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 14:25 monthly
drwxr-xr-x 2 dvgandich studsci    2048 мар  9 15:23 my_os
drwx--x--x 3 dvgandich studsci    2048 мар  9 15:39 play
-rw-r--r-- 1 dvgandich studsci   4995 фев 22 14:54 presentation.md
drwxr-xr-x 3 dvgandich root      2048 фев  4 20:29 public
lrwxr-xr-x 1 dvgandich root        18 мар  2 22:29 public_html -> public/public_html
-rw-r--r-- 1 dvgandich studsci 4251441 мар  9 14:44 report.pdf
drwxr-xr-x 3 dvgandich studsci    2048 мар  9 14:30 reports
drwxr-xr-x 4 dvgandich studsci    2048 мар  9 15:02 ski_places
drwxr-xr-x 2 dvgandich studsci    2048 сен 28 10:55 tmp
drwxr-xr-x 5 dvgandich studsci    2048 фев 22 16:18 work
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Видео
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Документы
drwxr-xr-x 3 dvgandich studsci    2048 ноя 16 11:39 Изображения
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Музыка
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Общедоступные
drwxr-xr-x 2 dvgandich studsci    2048 ноя  9 14:31 'Рабочий стол'
drwxr-xr-x 2 dvgandich studsci    2048 сен  7 2022 Шаблоны
```

Рис. 2.9: Пункт 4

5. Выполняем пункт 5 из текста ЛР, читаем map по командам mount, fsck, mkfs, kill

```

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|rslave|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 to the root of the filesystem on device.

    If only the directory or the device is given, for example:

        mount /dir

```

Рис. 2.10: 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 an 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

    16
        Usage or syntax error

    32
        Checking canceled by user request

    128
        Shared-library error
```

Рис. 2.11: 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.

    -V, --version
        Print version and exit. (Option -V will display version information only when it is the only parameter,
        otherwise it will work as --verbose.)

```

Рис. 2.12: 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
        List signal names in a nice table.

NOTES
    Your shell (command line interpreter) may have a built-in kill command. You may need to run the command described here as /bin/kill to solve the conflict.
```

Рис. 2.13: kill

```
dvgandich@dk4n64 ~ $ man mount
[1]+  Остановлен   man mount
dvgandich@dk4n64 ~ $ man fsck
[2]+  Остановлен   man fsck
dvgandich@dk4n64 ~ $ man mkfs
[3]+  Остановлен   man mkfs
dvgandich@dk4n64 ~ $ man kill
[4]+  Остановлен   man kill
```

Рис. 2.14: man

6. Краткая характеристика: -mount применяется для монтирования файловых систем, -fsck восстанавливает поврежденную файловую систему или проверяет на целостность, -mkfs создает новую файловую систему, -kill используется для принудительного завершения работы приложений.

3 Вывод

Мы повторили команды для взаимодействия с файлами и каталогами (`mv`, `cp`, `mkdir`), разобрались с тем, как работают и как изменять права доступа