

# **Отчет по лабораторной работе №5**

*Дисциплина: Операционные системы*

Кабанова Варвара Дмитриевна

# Содержание

1	Цель работы	6
2	Выполнение лабораторной работы	7
3	Выводы	19
4	Контрольные вопросы	20
	Список литературы	28

# Список иллюстраций

2.1	рис.1	7
2.2	рис.2	7
2.3	рис.3	7
2.4	рис.4	8
2.5	рис.5	8
2.6	рис.6	8
2.7	рис.7	8
2.8	рис.8	8
2.9	рис.9	9
2.10	рис.10	9
2.11	рис.11	9
2.12	рис.12	9
2.13	рис.13	9
2.14	рис.14	10
2.15	рис.15	10
2.16	рис.16	10
2.17	рис.17	10
2.18	рис.18	10
2.19	рис.19	10
2.20	рис.20	11
2.21	рис.21	11
2.22	рис.22	11
2.23	рис.23	11
2.24	рис.24	11
2.25	рис.25	12
2.26	рис.26	12
2.27	рис.27	12
2.28	рис.28	12
2.29	рис.29	12
2.30	рис.30	13
2.31	рис.31	13
2.32	рис.32	13
2.33	рис.33	14
2.34	рис.34	14
2.35	рис.35	14
2.36	рис.36	15
2.37	рис.37	15

2.38 рис.38 . . . . .	15
2.39 рис.39 . . . . .	15
2.40 рис.40 . . . . .	15
2.41 рис.41 . . . . .	16
2.42 рис.42 . . . . .	16
2.43 рис.43 . . . . .	16
2.44 рис.44 . . . . .	16
2.45 рис.45 . . . . .	16
2.46 рис.46 . . . . .	17
2.47 рис.47 . . . . .	17
2.48 рис.48 . . . . .	18
2.49 рис.49 . . . . .	18

## 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

Выполняю все примеры, описанные в лабораторной работе №5. Рассмотрим копирование файлов и каталогов. Выполним копирование файла в текущем каталоге. Копирую файл ~/abc1 в файл april и в файл may (рис.1).

```
vdkabanova@dk2n25 ~ $ cd
vdkabanova@dk2n25 ~ $ touch abc1
vdkabanova@dk2n25 ~ $ cp abc1 april
vdkabanova@dk2n25 ~ $ cp abc1 may
```

Рис. 2.1: рис.1

Выполним копирование нескольких файлов в каталог. Копирую файлы april и may в каталог monthly (рис.2).

```
vdkabanova@dk2n25 ~ $ mkdir monthly
vdkabanova@dk2n25 ~ $ cp april may monthly
```

Рис. 2.2: рис.2

Выполним копирование файлов в произвольном каталоге. Копирую файл monthly/may в файл с именем june (рис.3).

```
vdkabanova@dk2n25 ~ $ cp monthly/may monthly/june
vdkabanova@dk2n25 ~ $ ls monthly
april june may
```

Рис. 2.3: рис.3

Выполним копирование каталогов в текущем каталоге. Копирую каталог monthly в каталог monthly.00 (рис.4).

```
vdkabanova@dk2n25 ~ $ mkdir monthly.00  
vdkabanova@dk2n25 ~ $ cp -r monthly monthly.00
```

Рис. 2.4: рис.4

Выполним копирование каталогов в произвольном каталоге. Копирую каталог monthly.00 в каталог /tmp (рис.5).

```
vdkabanova@dk2n25 ~ $ cp -r monthly.00 /tmp
```

Рис. 2.5: рис.5

Рассмотрим перемещение и переименование файлов и каталогов. Выполним переименование файлов в текущем каталоге. Изменяю название файла april на july в домашнем каталоге (рис.6).

```
vdkabanova@dk2n25 ~ $ cd  
vdkabanova@dk2n25 ~ $ mv april july
```

Рис. 2.6: рис.6

Выполним перемещение файлов в другой каталог. Перемещаю файл july в каталог monthly.00 (рис.7).

```
vdkabanova@dk2n25 ~ $ mv july monthly.00  
vdkabanova@dk2n25 ~ $ ls monthly.00  
july  monthly
```

Рис. 2.7: рис.7

Выполним переименование каталогов в текущем каталоге. Переименовываю каталог monthly.00 в monthly.01 (рис.8).

```
vdkabanova@dk2n25 ~ $ mv monthly.00 monthly.01
```

Рис. 2.8: рис.8

Выполним перемещение каталога в другой каталог. Перемещаю каталог monthly.01 в каталог reports (рис.9).



```
vdkabanova@dk2n25 ~ $ mkdir reports
vdkabanova@dk2n25 ~ $ mv monthly.01 reports
```

Рис. 2.9: рис.9

Выполним переименование каталога, не являющегося текущим. Переименовываю каталог reports/monthly.01 в reports/monthly (рис.10).

```
vdkabanova@dk2n25 ~ $ mv reports/monthly.01 reports/monthly
vdkabanova@dk2n25 ~ $ cd
```

Рис. 2.10: рис.10

Рассмотрим изменение прав доступа. Создаю файл ~/may с правом выполнения для владельца (рис.11).

```
vdkabanova@dk2n25 ~ $ cd
vdkabanova@dk2n25 ~ $ touch may
vdkabanova@dk2n25 ~ $ ls -l may
-rw-r--r-- 1 vdkabanova studsci 0 мая  4 17:32 may
```

Рис. 2.11: рис.11

Лишаю владельца файла ~/may права на выполнение (рис.12).

```
vdkabanova@dk2n25 ~ $ chmod u-x may
vdkabanova@dk2n25 ~ $ ls -l may
-rwxr--r-- 1 vdkabanova studsci 0 мая  4 17:32 may
```

Рис. 2.12: рис.12

Создаю каталог monthly с запретом на чтение для членов группы и всех остальных пользователей (рис. 13).

```
vdkabanova@dk2n25 ~ $ chmod u-x may
vdkabanova@dk2n25 ~ $ ls -l may
-rwxr--r-- 1 vdkabanova studsci 0 мая  4 17:32 may
```

Рис. 2.13: рис.13

Создаю файл ~/abc1 с правом записи для членов группы (рис. 14-16).

```

vdkabanova@dk2n25 ~ $ cd
vdkabanova@dk2n25 ~ $ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует

```

Рис. 2.14: рис.14

```

vdkabanova@dk2n25 ~ $ cd
vdkabanova@dk2n25 ~ $ touch abc1
vdkabanova@dk2n25 ~ $ chmod g+w abc1

```

Рис. 2.15: рис.15

```

vdkabanova@dk2n25 ~ $ chmod go-r monthly
vdkabanova@dk2n25 ~ $ ls -l
итого 37
-rw-r--r-- 1 vdkabanova studsci  0 мая  4 17:25 abc1
drwxr-xr-x 2 vdkabanova studsci 2048 апр 28 15:02 bin
drwxr-xr-x 3 vdkabanova studsci 2048 сен  2 2021 GNUstep
-rw-r--r-- 1 vdkabanova studsci  0 мая  4 17:32 may
drwx--x--x 2 vdkabanova studsci 2048 мая  4 17:27 monthly
drwxr-xr-x 3 vdkabanova studsci 2048 сен  1 2021 public
lrwxr-xr-x 1 vdkabanova root    18 апр 14 23:34 public_html -> public/public_html
drwxr-xr-x 3 vdkabanova studsci 2048 мая  4 17:30 reports
drwxr-xr-x 2 vdkabanova studsci 2048 ноя 10 18:47 tmp
drwxr-xr-x 3 vdkabanova studsci 2048 апр 21 16:13 work
drwxr-xr-x 2 vdkabanova studsci 2048 сен  2 2021 Документы
drwxr-xr-x 3 vdkabanova studsci 2048 апр 28 15:02 Загрузки
drwxr-xr-x 2 vdkabanova studsci 10240 апр 29 13:18 Изображения
drwxr-xr-x 3 vdkabanova studsci 2048 окт  8 2021 Музыка
drwxr-xr-x 2 vdkabanova studsci 2048 сен  2 2021 Общедоступные
drwxr-xr-x 2 vdkabanova studsci 2048 окт 13 2021 'Рабочий стол'
drwxr-xr-x 2 vdkabanova studsci 2048 сен  2 2021 Шаблоны

```

Рис. 2.16: рис.16

Копирую файл /usr/include/sys/io.h в домашний каталог (рис. 17) и называю его equipment (рис. 18).

```

vdkabanova@dk2n25 ~ $ cp /usr/include/sys/io.h ~

```

Рис. 2.17: рис.17

```

vdkabanova@dk2n25 ~ $ cp /usr/include/sys/io.h equipment
vdkabanova@dk2n25 ~ $ ls
abc1      equipment  may        public_html  work        Изображения  'Рабочий стол'
bin       GNUstep    monthly    reports      Документы    Музыка        Шаблоны
'~equipment' io.h       public     tmp          Загрузки     Общедоступные

```

Рис. 2.18: рис.18

В домашнем каталоге создаю директорию ~/ski.places (рис. 19).

```

vdkabanova@dk2n25 ~ $ mkdir ski.places

```

Рис. 2.19: рис.19

Перемещаю файл equipment в каталог ~/ski.plases (рис.20).

```
vdkabanova@dk2n25 ~ $ mv equipment ski.plases
```

Рис. 2.20: рис.20

Переименовываю файл ~/ski.plases/equipment в ~/ski.plases/equiplist (рис.21).

```
vdkabanova@dk2n25 ~ $ mv ski.plases/equipment ski.plases/equiplist
```

Рис. 2.21: рис.21

Создаю в домашнем каталоге файл abc1, копирую его в каталог ~/ski.plases, называю его equiplist2 (рис. 22).

```
vdkabanova@dk2n25 ~ $ touch abc1  
vdkabanova@dk2n25 ~ $ cp abc1 ski.plases  
vdkabanova@dk2n25 ~ $ mv ski.plases/abc1 ski.plases/equiplist2
```

Рис. 2.22: рис.22

Создаю каталог с именем equipment в каталоге ~/ski.plases (рис.23).

```
vdkabanova@dk2n25 ~ $ cd ski.plases  
vdkabanova@dk2n25 ~/ski.plases $ mkdir equipment  
vdkabanova@dk2n25 ~/ski.plases $ ls  
equiplist  equiplist2  equipment
```

Рис. 2.23: рис.23

Перемещаю файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment (рис. 24).

```
vdkabanova@dk2n25 ~ $ mv ski.plases/equiplist ski.plases/equipment  
vdkabanova@dk2n25 ~ $ mv ski.plases/equiplist2 ski.plases/equipment
```

Рис. 2.24: рис.24

Создаю и перемещаю каталог ~/newdir в каталог ~/ski.plases и называю его plans (рис. 25-26).

```
vdkabanova@dk2n25 ~ $ mkdir newdir
vdkabanova@dk2n25 ~ $ mv newdir plans
vdkabanova@dk2n25 ~ $ mv plans ski.plases
```

Рис. 2.25: рис.25

```
vdkabanova@dk2n25 ~ $ ls ski.plases
equipment  plans
```

Рис. 2.26: рис.26

Создаю нужные каталоги и файлы (рис. 27).

```
vdkabanova@dk2n25 ~ $ mkdir australia
vdkabanova@dk2n25 ~ $ mkdir play
vdkabanova@dk2n25 ~ $ touch my_os
vdkabanova@dk2n25 ~ $ touch feathers
```

Рис. 2.27: рис.27

Определяю опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

- `drwxr-r- ... australia` (рис.28)

```
vdkabanova@dk2n25 ~ $ chmod u+rwx australia
vdkabanova@dk2n25 ~ $ chmod g+r australia
vdkabanova@dk2n25 ~ $ chmod o+r australia
```

Рис. 2.28: рис.28

- `drwx-x-x ... play` (рис. 29)

```
vdkabanova@dk2n25 ~ $ chmod u+rwx play
vdkabanova@dk2n25 ~ $ chmod g+x play
vdkabanova@dk2n25 ~ $ chmod o+x play
```

Рис. 2.29: рис.29

- -r-xr-r- ... my\_os (рис. 30)

```
vdkabanova@dk2n25 ~ $ chmod u+r my_os
vdkabanova@dk2n25 ~ $ chmod g+rx my_os
vdkabanova@dk2n25 ~ $ chmod o+r my_os
```

Рис. 2.30: рис.30

- -rw-rw-r- ... feathers (рис. 31)

```
vdkabanova@dk2n25 ~ $ chmod u+rw feathers
vdkabanova@dk2n25 ~ $ chmod g+rw feathers
vdkabanova@dk2n25 ~ $ chmod o+r feathers
```

Рис. 2.31: рис.31

Просматриваю содержимое домашней папки (рис. 32).

```
vdkabanova@dk2n25 ~ $ ls -l
итого 53
-rw-rw-r-- 1 vdkabanova studsci  0 мая 4 17:47 abc1
drwxr--r-- 2 vdkabanova studsci 2048 мая 4 17:58 australia
drwxr-xr-x 2 vdkabanova studsci 2048 апр 28 15:02 bin
-rw-r--r-- 1 vdkabanova studsci 5086 мая 4 17:44 'equipment'
-rw-rw-r-- 1 vdkabanova studsci  0 мая 4 17:59 feathers
drwxr-xr-x 3 vdkabanova studsci 2048 сен 2 2021 GNUstep
-rw-r--r-- 1 vdkabanova studsci 5086 мая 4 17:43 io.h
-rw-r--r-- 1 vdkabanova studsci  0 мая 4 17:32 may
drwx--x--x 2 vdkabanova studsci 2048 мая 4 17:27 monthly
-rw-r-xr-- 1 vdkabanova studsci  0 мая 4 17:58 my_os
drwx--x--x 2 vdkabanova studsci 2048 мая 4 17:58 play
drwxr-xr-x 3 vdkabanova studsci 2048 сен 1 2021 public
lrwxr-xr-x 1 vdkabanova root    18 апр 14 23:34 public_html -> public/public_html
drwxr-xr-x 3 vdkabanova studsci 2048 мая 4 17:30 reports
drwxr-xr-x 4 vdkabanova studsci 2048 мая 4 17:57 ski.places
drwxr-xr-x 2 vdkabanova studsci 2048 ноя 10 18:47 tmp
drwxr-xr-x 3 vdkabanova studsci 2048 апр 21 16:13 work
drwxr-xr-x 2 vdkabanova studsci 2048 сен 2 2021 Документы
drwxr-xr-x 3 vdkabanova studsci 2048 апр 28 15:02 Загрузки
drwxr-xr-x 2 vdkabanova studsci 10240 мая 4 18:09 Изображения
drwxr-xr-x 3 vdkabanova studsci 2048 окт 8 2021 Музыка
drwxr-xr-x 2 vdkabanova studsci 2048 сен 2 2021 Общедоступные
drwxr-xr-x 2 vdkabanova studsci 2048 окт 13 2021 'Рабочий стол'
drwxr-xr-x 2 vdkabanova studsci 2048 сен 2 2021 Шаблоны
```

Рис. 2.32: рис.32

Просматриваю содержимое файла /etc/passwd (рис. 33-34).

```
vdkabanova@dk2n25 ~ $ cat /etc/passwd
root:x:0:0:System user; root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/false
daemon:x:2:2:daemon:/sbin:/bin/false
adm:x:3:4:adm:/var/adm:/bin/false
lp:x:4:7:lp:/var/spool/lpd:/bin/false
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:Mail program user:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/bin/false
uucp:x:10:14:uucp:/var/spool/uucppublic:/bin/false
operator:x:11:0:operator:/root:/bin/bash
man:x:13:15:System user; man:/dev/null:/sbin/nologin
postmaster:x:14:12:Postmaster user:/var/spool/mail:/sbin/nologin
cron:x:16:16:A user for sys-process/cronbase:/var/spool/cron:/sbin/nologin
ftp:x:21:21::/home/ftp:/bin/false
sshd:x:22:22>User for ssh:/var/empty:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/bin/false
squid:x:31:31:Squid:/var/cache/squid:/bin/false
gdm:x:32:32>User for running GDM:/var/lib/gdm:/sbin/nologin
xfs:x:33:33:X Font Server:/etc/X11/fs:/bin/false
games:x:35:35:games:/usr/games:/bin/bash
named:x:40:40:bind:/var/bind:/bin/false
mysql:x:60:60:MySQL program user:/dev/null:/sbin/nologin
postgres:x:70:70:PostgreSQL program user:/var/lib/postgresql:/bin/sh
```

Рис. 2.33: рис.33

```
pcscd:x:124:999:A user for pcsc-lite:/dev/null:/sbin/nologin
apache:x:81:81:System user; apache:/var/www:/sbin/nologin
colord:x:125:995>User for running the colord service:/var/lib/colord:/sbin/nologin
munge:x:126:994:A user for sys-auth/munge:/var/lib/munge:/sbin/nologin
slurm:x:500:993:added by portage for slurm:/var/spool/slurm:/sbin/nologin
qemu:x:77:997:System user; qemu:/dev/null:/sbin/nologin
polkitd:x:127:992:System user; polkitd:/var/lib/polkit-1:/sbin/nologin
radvd:x:128:991:A user for the Router Advertisement Daemon:/dev/null:/sbin/nologin
dnsmasq:x:129:990>User for net-dns/dnsmasq:/dev/null:/sbin/nologin
saned:x:130:442>User for media-gfx/sane-backends:/dev/null:/sbin/nologin
aspnet:x:131:989:added by portage for xsp:/tmp:/sbin/nologin
systemd-journal-gateway:x:132:987:added by portage for systemd:/dev/null:/sbin/nologin
puppet:x:133:986:added by portage for puppet:/var/lib/puppet:/sbin/nologin
systemd-bus-proxy:x:134:985:added by portage for systemd:/dev/null:/sbin/nologin
systemd-network:x:135:984:System user; systemd-network:/dev/null:/sbin/nologin
systemd-resolve:x:136:983:System user; systemd-resolve:/dev/null:/sbin/nologin
systemd-timesync:x:137:982:System user; systemd-timesync:/dev/null:/sbin/nologin
geoclue:x:138:981>User for running the GeoClue D-Bus geolocation service:/var/lib/geoclue:/sbin/nologin
systemd-journal-remote:x:139:979:System user; systemd-journal-remote:/dev/null:/sbin/nologin
systemd-journal-upload:x:140:978:added by portage for systemd:/dev/null:/sbin/nologin
sddm:x:141:977:added by portage for sddm:/var/lib/sddm:/sbin/nologin
localadmin:x:1000:100::/home/localadmin:/bin/bash
systemd-coredump:x:142:976:System user; systemd-coredump:/dev/null:/sbin/nologin
spamd:x:143:143>User for the SpamAssassin daemon:/var/lib/spamd:/sbin/nologin
nslcd:x:144:973>User for nss-pam-ldapd:/dev/null:/sbin/nologin
epmd:x:335:335>User for Erlang Portmapper Daemon:/dev/null:/sbin/nologin
xrootd:x:998:971>User for the XRootD server:/dev/null:/sbin/nologin
tss:x:59:59:Trusted Software Stack for TPMs user:/dev/null:/sbin/nologin
svn:x:399:399:System user; svn:/dev/null:/sbin/nologin
systemd-oom:x:198:198:System user; systemd-oom:/dev/null:/sbin/nologin
firebird:x:450:450:added by portage for firebird:/usr/lib64/firebird:/bin/sh
sndiod:x:461:18:Daemon user for sndio:/dev/null:/sbin/nologin
```

Рис. 2.34: рис.34

Копирую файл ~/feathers в файл ~/file.old (рис. 35).

```
vdkabanova@dk2n25 ~ $ cp feathers file.old
```

Рис. 2.35: рис.35

Перемещаю файл ~/file.old в каталог ~/play (рис. 36).

```
vdkabanova@dk2n25 ~ $ mv file.old play
```

Рис. 2.36: рис.36

Копирую каталог ~/play в каталог ~/fun. Перемещаю каталог ~/fun в каталог ~/play и называю его games (рис. 37).

```
vdkabanova@dk2n25 ~ $ cp -r play fun
vdkabanova@dk2n25 ~ $ ls
abcl    '~equipment'  games    may    play    reports  work    Изображения  'Рабочий стол'
australia  feathers      GNUstep  monthly  public  ski_places  Документы  Музыка  'Шаблоны'
bin      fun          io.h    my_os   public_html  tmp      Загрузки    Общедоступные
vdkabanova@dk2n25 ~ $ ls fun
play
vdkabanova@dk2n25 ~ $ mv fun play
vdkabanova@dk2n25 ~ $ mv play/fun play/games
vdkabanova@dk2n25 ~ $ ls play
file.old  games
```

Рис. 2.37: рис.37

Лишите владельца файла ~/feathers права на чтение (рис. 38).

```
vdkabanova@dk2n25 ~ $ chmod u-r feathers
```

Рис. 2.38: рис.38

Пытаясь просмотреть файл ~/feathers командой cat, вижу, что выскакивает уведомление “отказано в доступе” (рис. 39)

```
vdkabanova@dk2n25 ~ $ cat feathers
cat: feathers: Отказано в доступе
```

Рис. 2.39: рис.39

Пытаясь скопировать файл ~/feathers, вижу, что выскакивает уведомление “отказано в доступе” (рис. 40)

```
vdkabanova@dk2n25 ~ $ cp feathers play
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
```

Рис. 2.40: рис.40

Даю владельцу файла ~/feathers право на чтение (рис. 41).

```
vdkabanova@dk2n25 ~ $ chmod u+r feathers
```

Рис. 2.41: рис.41

Лишаю владельца каталога ~/play права на выполнение (рис. 42).

```
vdkabanova@dk2n25 ~ $ chmod u-x play
```

Рис. 2.42: рис.42

Несмотря на то, что мы лишили владельца каталога прав на выполнение, переход в катлог ~/play все равно возможен (рис. 43).

```
vdkabanova@dk2n25 ~ $ cd play  
vdkabanova@dk2n25 ~/play $
```

Рис. 2.43: рис.43

Даю владельцу каталога ~/play право на выполнение (рис. 44).

```
vdkabanova@dk2n25 ~ $ chmod u+x play
```

Рис. 2.44: рис.44

Прочитываю man по командам (рис. 45)

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

Рис. 2.45: рис.45

- mount (рис. 46)



```

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:

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

```

Рис. 2.46: рис.46

- fsck (рис. 47)

```

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/cd0c2), a mount point (e.g., /, /usr, /home), or an filesystem label or UUID specifier
    (e.g., UUID=8868abf6-8865-4ab3-98b8-bfc2405f77bd 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

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

```

Рис. 2.47: рис.47

- mkfs (рис. 48)

```

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.

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

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

```

Рис. 2.48: рис.48

- kill (рис. 49)

```

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.

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

```

Рис. 2.49: рис.49

## 3 Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов; 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

## 4 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу.
2. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.
3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?
4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?
5. Как создаётся файловая система?
6. Дайте характеристику командам для просмотра текстовых файлов.
7. Приведите основные возможности команды `cp` в Linux.
8. Приведите основные возможности команды `mv` в Linux.
9. Что такое права доступа? Как они могут быть изменены?
- 1) Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «`df-Th`». Из рисунка видно, что 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 предназначена 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 «`extent file 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: 16 TB; 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. Адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога. Только пользователь `root` имеет право читать и изменять файлы в этом каталоге. `/BIN` – бинарные файлы пользователя. Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления. `/SBIN` – системные исполняемые файлы. Также как и `/bin`, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог `/usr`. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя. `/ETC` – конфигурационные файлы. В этой папке содержатся конфигурационные файлы всех программ, установленных в системе. Кроме конфигурационных файлов, в системе инициализации `Init Scripts`, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ. `/DEV` – файлы устройств. В Linux все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны,

камеры –это просто файлы в каталоге /dev/. Выполняется сканирование всех подключенных устройств и создание для них специальных файлов. “/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. Также здесь можно найти информацию об использовании системных ресурсов. “/VAR” – переменные файлы. Название каталога “/var” говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кешы, базы данных и так далее. “/TMP” – временные файлы В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию. “/USR” – программы пользователя Это самый большой каталог с большим количеством функций. 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 проприетарные 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” -процессыКаталог,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) Целостность файловой системы может быть нарушена из-за перебоев в питании, неполадок в оборудовании или из-за некорректного/внезапного выключения компьютера. 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-Е–показыватьсимвол \$ 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 командecat, новыводит еще иномера строк встолбце слева. less. Существенно более развитая командадля 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 синтаксисукоманды 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, не добавляя 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: = устано-

вить право-лишить права+ датьправог чтениенw записьх выполнениенu (user)  
владелец файлаg (group) группа, к которой принадлежит владелец файлаo  
(others) все остальные.

## **Список литературы**