

# Отчет по лабораторной работе №6

## Дисциплина: Операционные системы

Калистратова Ксения Евгеньевна

### Содержание

Цель работы .....	1
Задачи.....	1
Выполнение лабораторной работы .....	1
Контрольные вопросы .....	13
Выводы .....	20

### 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.

### Задачи

1. Изучить команды для работы с файлами и каталогами.
2. Изучить команды для копирования файлов и каталогов.
3. Изучить команды для перемещения и переименования файлов и каталогов.
4. Изучить права доступа файлов или каталогов.
5. В ходе работы использовать эти команды и интерпретировать их вывод.
6. Выполнить отчет.

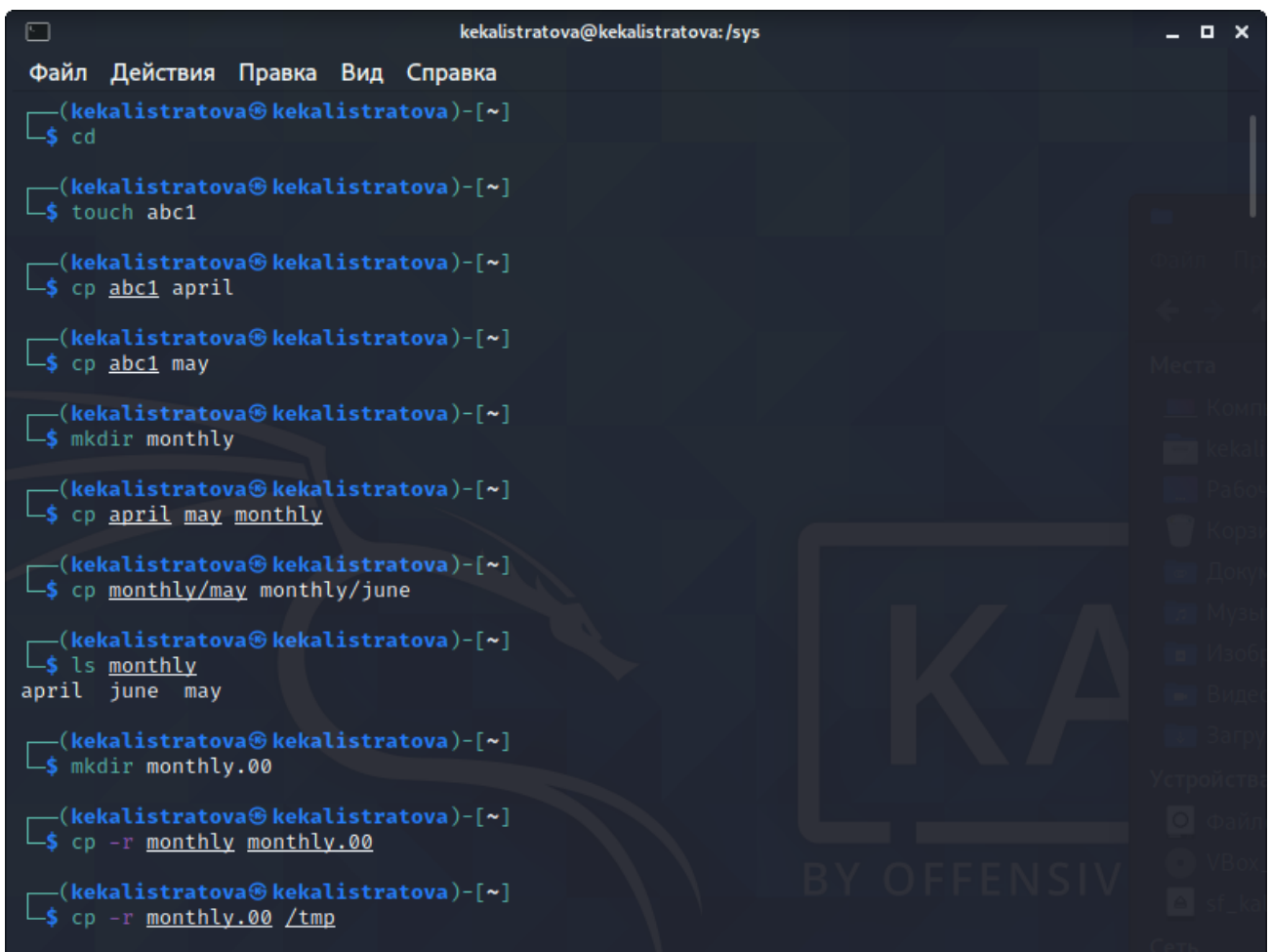
### Выполнение лабораторной работы

- 1) Для начала выполним примеры, описанные в первой части описания лабораторной работы.

Рисунок 1(рис. 1):

1. Скопируем файл ~/abc1 в файл april и в файл may. Для этого создадим файл abc1, используя команду «touch abc1», далее осуществим копирование с помощью команд «cp abc1 april» и «cp abc1 may».

2. Скопируем файлы `april` и `may` в каталог `monthly`, используя команды «`mkdir monthly`» – для создания каталога `monthly` и «`cp april may monthly`» – для копирования.
3. Скопируем файл `monthly/may` в файл с именем `june`. Выполним команды «`cp monthly/may monthly/june`» и «`ls monthly`» (для просмотра содержимого каталога).
4. Скопируем каталог `monthly` в каталог `monthly.00`. Для этого создадим каталог `monthly.00` командой «`mkdir monthly.00`» и осуществим копирование, используя команду «`cp -r monthly monthly.00`» (команда `cp` с опцией `r` (recursive) позволяет копировать каталоги вместе с входящими в них файлами и каталогами).
5. Скопируем каталог `monthly.00` в каталог `/tmp`, используя команду «`cp -r monthly.00 /tmp`».



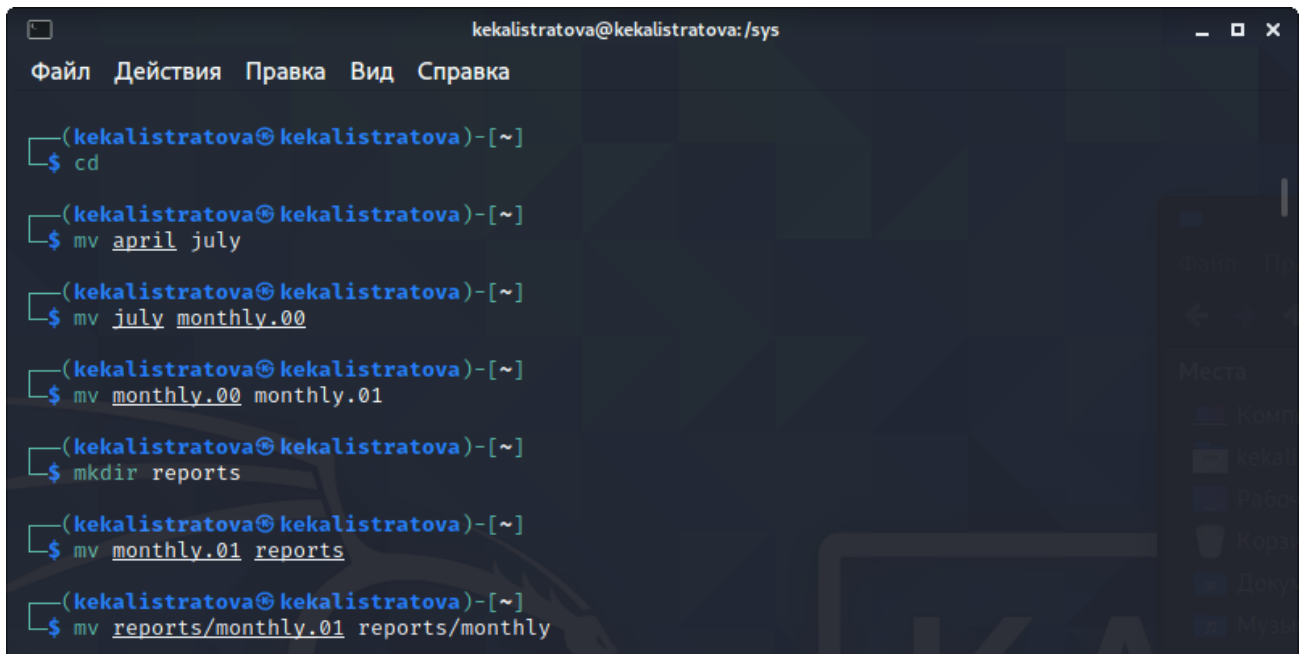
```
kekalistratova@kekalistratova: /sys
Файл Действия Правка Вид Справка
(kekalistratova@kekalistratova)-[~]
$ cd
(kekalistratova@kekalistratova)-[~]
$ touch abc1
(kekalistratova@kekalistratova)-[~]
$ cp abc1 april
(kekalistratova@kekalistratova)-[~]
$ cp abc1 may
(kekalistratova@kekalistratova)-[~]
$ mkdir monthly
(kekalistratova@kekalistratova)-[~]
$ cp april may monthly
(kekalistratova@kekalistratova)-[~]
$ cp monthly/may monthly/june
(kekalistratova@kekalistratova)-[~]
$ ls monthly
april june may
(kekalistratova@kekalistratova)-[~]
$ mkdir monthly.00
(kekalistratova@kekalistratova)-[~]
$ cp -r monthly monthly.00
(kekalistratova@kekalistratova)-[~]
$ cp -r monthly.00 /tmp
```

Figure 1: Копирование файлов и каталогов

Рисунок 2(рис. 2):

1. Изменим название файла `april` на `july` в домашнем каталоге, используя команду «`mv april july`».

2. Переместим файл `july` в каталог `monthly.00` с помощью команды «`mv july monthly.00`». Проверим результат командой «`ls monthly.00`».
3. Переименуем каталог `monthly.00` в `monthly.01`, используя команду «`mv monthly.00 monthly.01`».
4. Переместим каталог `monthly.01` в каталог `reports`. Для этого создадим каталог `reports` с помощью команды «`mkdir reports`» и выполним перемещение командой «`mv monthly.01 reports`».
5. Переименуем каталог `reports/monthly.01` в `reports/monthly` командой «`mv reports/monthly.01 reports/monthly`».



```
kekalistratova@kekalistratova:/sys
Файл Действия Правка Вид Справка

(kekalistratova@kekalistratova)-[~]
$ cd

(kekalistratova@kekalistratova)-[~]
$ mv april july

(kekalistratova@kekalistratova)-[~]
$ mv july monthly.00

(kekalistratova@kekalistratova)-[~]
$ mv monthly.00 monthly.01

(kekalistratova@kekalistratova)-[~]
$ mkdir reports

(kekalistratova@kekalistratova)-[~]
$ mv monthly.01 reports

(kekalistratova@kekalistratova)-[~]
$ mv reports/monthly.01 reports/monthly
```

Figure 2: Перемещение и переименование файлов и каталогов

Рисунок 3(рис. 3):

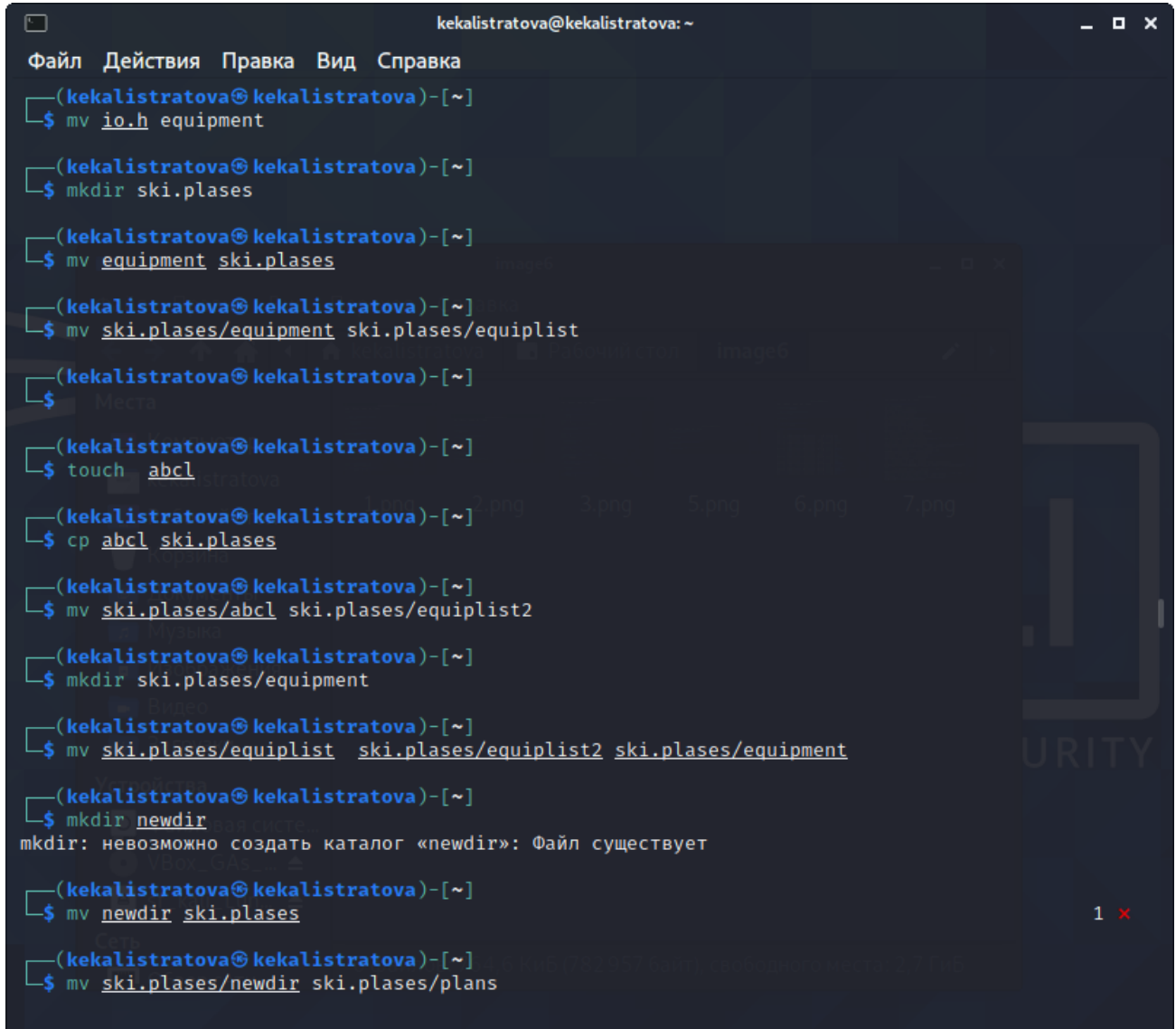
1. Создадим файл `~/may` с правом выполнения для владельца. Для этого выполним следующие команды: «`touch may`» (создание файла), «`ls -l may`» (просмотр сведений о файле), «`chmod u+x may`» (изменение прав), «`ls -l may`».
2. Лишаем владельца файла `~/may` права на выполнение, используя команды: «`chmod u-x may`» (изменение прав), «`ls -l may`» (просмотр сведений о файле).
3. Создаем каталог `monthly` с запретом на чтение для членов группы и всех остальных пользователей. Выполняем команды: «`mkdir monthly`» (создание каталога), «`chmod go-r monthly`»(изменение прав).
4. Создаем файл `~/abcl` с правом записи для членов группы, используя команды: «`touch abcl`» (создание файла), «`chmod g+w abcl`» (изменение прав).

```
kekalistratova@kekalistratova:/sys
Файл Действия Правка Вид Справка
└─$ cd
└─(kekalistratova@kekalistratova)-[~]
└─$
└─(kekalistratova@kekalistratova)-[~]
└─$ touch may
└─(kekalistratova@kekalistratova)-[~]
└─$ ls -l may
-rw-r--r-- 1 kekalistratova kekalistratova 0 мая 15 12:37 may
└─(kekalistratova@kekalistratova)-[~]
└─$ chmod u+x may
└─(kekalistratova@kekalistratova)-[~]
└─$ ls -l may
-rwxr--r-- 1 kekalistratova kekalistratova 0 мая 15 12:37 may
└─(kekalistratova@kekalistratova)-[~]
└─$ chmod u-x may
└─(kekalistratova@kekalistratova)-[~]
└─$ ls -l may
-rw-r--r-- 1 kekalistratova kekalistratova 0 мая 15 12:37 may
└─(kekalistratova@kekalistratova)-[~]
└─$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
└─(kekalistratova@kekalistratova)-[~]
└─$ chmod go-r monthly
└─(kekalistratova@kekalistratova)-[~]
└─$ touch abcl
└─(kekalistratova@kekalistratova)-[~]
└─$ chmod g+w abcl
```

Figure 3: Изменение прав доступа

- 2) Выполняем следующие действия (рис. 4):
  1. Копируем файл /usr/include/sys/io.h в домашний каталог (команда «cp /usr/include/sys/io.h ~») и называем его equipment (команда «mv io.h equipment»).
  2. В домашнем каталоге создаем директорию ~/ski.plases (команда «mkdir ski.plases»).
  3. Перемещаем файл equipment в каталог ~/ski.plases (команда «mv equipmentski.plases»).
  4. Переименовываем файл ~/ski.plases/equipment в ~/ski.plases/equiplist (команда «mv ski.plases/equipment ski.plases/equiplist»).
  5. Создаем в домашнем каталоге файла abcl (команда «touch abcl») и копируем его в каталог ~/ski.plases (команда «cp abcl ski.plases»), называем его equiplist2 (команда «mv ski.plases/abcl ski.plases/equiplist2»).

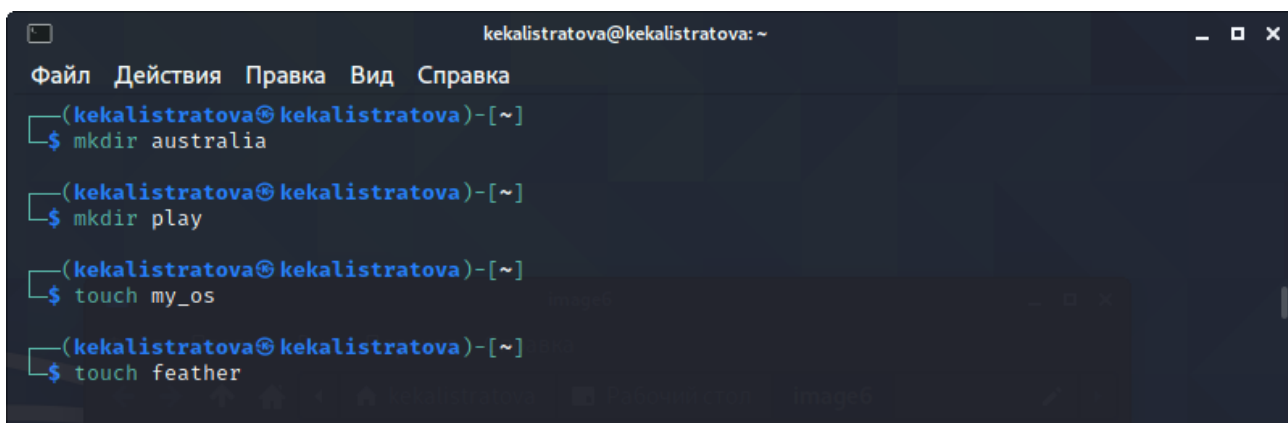
6. Создаем каталог с именем equipment в каталоге ~/ski.plases (команда «mkdir ski.plases/equipment»).
7. Перемещаем файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment (команда «mv ski.plases/equiplist ski.plases/equipment»).
8. Создаем (команда «mkdir newdir») и перемещаем каталог ~/newdir в каталог~/ski.plases (команда «mv newdir ski.plases») и называем его plans (команда «mv ski.plases/newdir ski.plases/plans»)



```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
$ mv io.h equipment  
$ mkdir ski.plases  
$ mv equipment ski.plases  
$ mv ski.plases/equipment ski.plases/equiplist  
$ touch abcl  
$ cp abcl ski.plases  
$ mv ski.plases/abcl ski.plases/equiplist2  
$ mkdir ski.plases/equipment  
$ mv ski.plases/equiplist ski.plases/equiplist2 ski.plases/equipment  
$ mkdir newdir  
mkdir: невозможно создать каталог «newdir»: Файл существует  
$ mv newdir ski.plases  
$ mv ski.plases/newdir ski.plases/plans
```

Figure 4: Создаем, перемещаем и переименовываем файлы и каталоги

- 3) Определяем опции команды chmod, необходимые для того, чтобы присвоить соответствующим файлам выделенные права доступа, считая, что в начале таких прав нет. Предварительно создаем необходимые файлы, используя команды: «mkdir australia», «mkdir play», «touch my\_os», «touch feathers». (рис. 5)

A screenshot of a terminal window with a dark background. The window title is 'kekalistratova@kekalistratova: ~'. At the top, there is a menu bar with the items 'Файл', 'Действия', 'Правка', 'Вид', and 'Справка'. The terminal shows four commands being executed in sequence, each preceded by a prompt '(kekalistratova@kekalistratova)-[~] \$'. The commands are: 'mkdir australia', 'mkdir play', 'touch my\_os', and 'touch feather'. Each command is followed by a new prompt line.

```
kekalistratova@kekalistratova: ~
Файл Действия Правка Вид Справка
(kekalistratova@kekalistratova)-[~]
$ mkdir australia
(kekalistratova@kekalistratova)-[~]
$ mkdir play
(kekalistratova@kekalistratova)-[~]
$ touch my_os
(kekalistratova@kekalistratova)-[~]
$ touch feather
```

Figure 5: Создание необходимых файлов

1. drwxr-r-... australia: команда «chmod 744 australia» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только чтение).
2. drwx-x-x ... play: команда «chmod 711 play» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только выполнение).
3. -r-xr-r-... my\_os: команды «chmod 544 my\_os» (это файл, владелец имеет право на чтение и выполнение, группа владельца и остальные – только чтение).
4. -rw-rw-r-... feathers: команды «chmod 664 feathers» (это файл, владелец и группа владельца имеют право на чтение и запись, остальные – только чтение)

Командой «ls-l» проверяем правильность выполненных действий. (рис. 6)

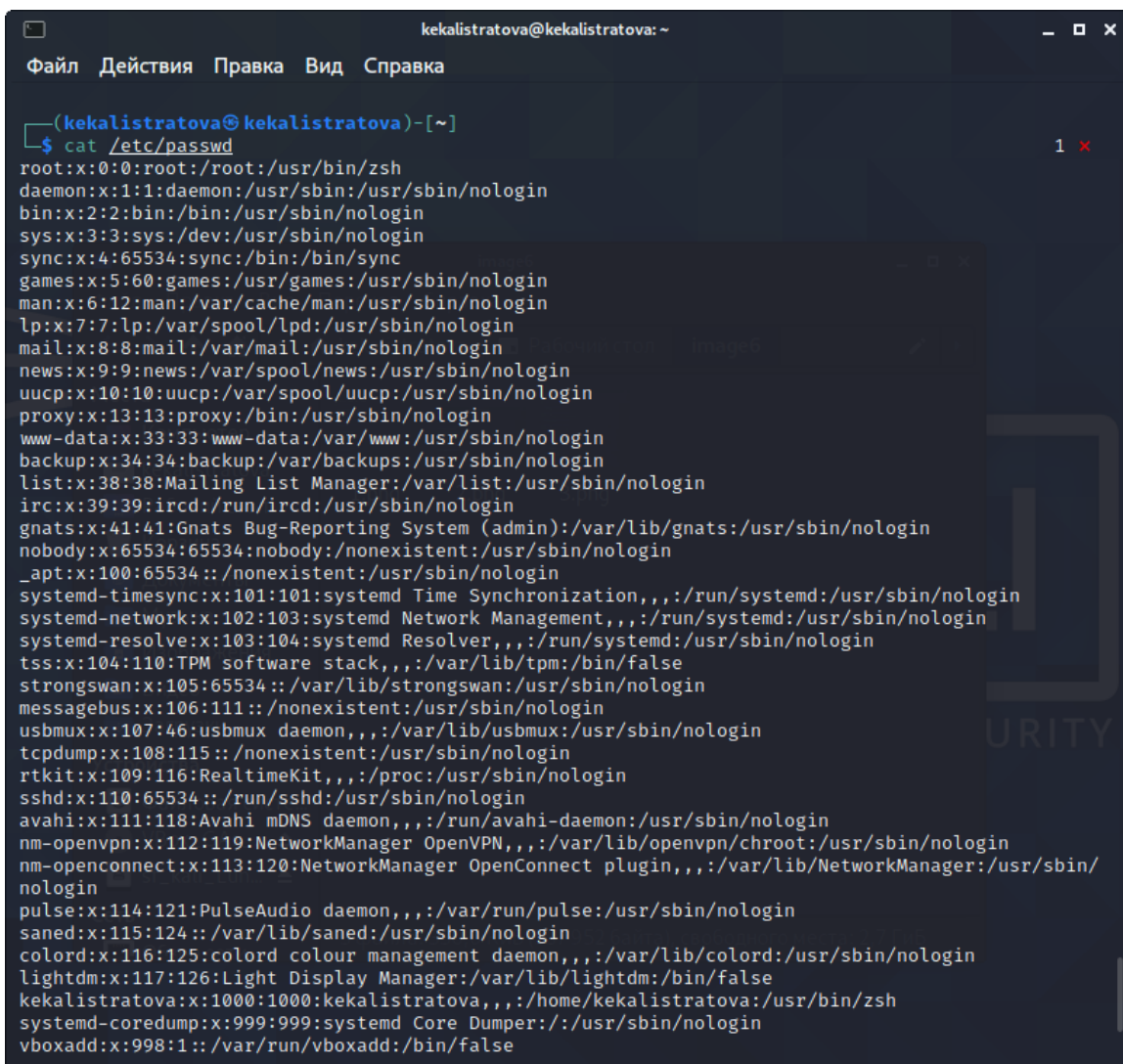
```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
(kekalistratova@kekalistratova)-[~]  
$ chmod 744 australia  
(kekalistratova@kekalistratova)-[~]  
$ chmod 711 play  
(kekalistratova@kekalistratova)-[~]  
$ chmod 544 my_os  
(kekalistratova@kekalistratova)-[~]  
$ chmod 664 feather  
(kekalistratova@kekalistratova)-[~]  
$ ls -l  
итого 76  
-rw-r--r-- 1 kekalistratova kekalistratova 0 мая 15 12:29 abc1  
-rw-r--r-- 1 kekalistratova kekalistratova 0 мая 15 12:29 abc11  
-rw-rw-r-- 1 kekalistratova kekalistratova 0 мая 16 14:45 abcl  
drwxr--r-- 2 kekalistratova kekalistratova 4096 мая 16 14:50 australia  
-rw-r--r-- 1 kekalistratova kekalistratova 4546 мая 15 13:30 clif.h  
-rw-rw-r-- 1 kekalistratova kekalistratova 0 мая 16 14:51 feather  
-rw-r--r-- 1 kekalistratova kekalistratova 0 мая 15 12:37 may  
drwx--x--x 2 kekalistratova kekalistratova 4096 мая 15 12:30 monthly  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 мая 12 16:12 morefun1  
-r-xr--r-- 1 kekalistratova kekalistratova 0 мая 16 14:51 my_os  
drwx--x--x 2 kekalistratova kekalistratova 4096 мая 16 14:50 play  
drwxr-xr-x 3 kekalistratova kekalistratova 4096 мая 15 12:36 reports  
drwxr-xr-x 4 kekalistratova kekalistratova 4096 мая 16 14:49 ski.places  
-rw-r--r-- 1 kekalistratova kekalistratova 11918 мая 15 13:34 sudo_plugin.h  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 26 23:49 Видео  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 26 23:49 Документы  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 мая 14 18:47 Загрузки  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 27 00:04 Изображения  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 26 23:49 Музыка  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 26 23:49 Общедоступные  
drwxr-xr-x 8 kekalistratova kekalistratova 4096 мая 16 14:30 'Рабочий стол'  
drwxr-xr-x 2 kekalistratova kekalistratova 4096 апр 26 23:49 Шаблоны
```

Figure 6: Опции команды chmod

- 4) Выполняем следующие действия (рис. 7) (рис. 8):
  1. Просмотрим содержимое файла /etc/passwd (команда «cat /etc/passwd»).
  2. Копируем файл ~/feathers в файл ~/file.old (команда «cp feathers file.old»).
  3. Переместим файл ~/file.old в каталог ~/play (команда «mv file.old play»).
  4. Скопируем каталог ~/play в каталог ~/fun (команда «cp -r play fun»).
  5. Переместим каталог ~/fun в каталог ~/play (команда «mv fun play») и назовем его games (команда «mv play/fun play/games»).
  6. Лишим владельца файла ~/feathers права на чтение (команда «chmod u-rfeathers»).
  7. Если мы попытаемся просмотреть файл ~/feathers командой cat, то получим отказ в доступе, т. к. в предыдущем пункте лишили владельца права на чтение данного файла.



8. Если мы попытаемся скопировать файл ~/feathers, например, в каталог monthly, то получим отказ в доступе, по причине, описанной в предыдущем пункте.
9. Дадим владельцу файла ~/feathers право на чтение (команда «chmod u+r feathers»).
10. Лишим владельца каталога ~/play права на выполнение (команда «chmod u-x play»).
11. Перейдем в каталог ~/play (команда «cd play»). Мы не получили отказ в доступе несмотря на то, что в предыдущем пункте лишили владельца права на выполнение данного каталога.
12. Дадим владельцу каталога ~/play право на выполнение (команда «chmod u+x play»).



The screenshot shows a terminal window with the title bar 'kekalistratova@kekalistratova: ~'. The terminal prompt is '(kekalistratova@kekalistratova)-[~]'. The user has entered the command 'cat /etc/passwd', and the output is displayed. The output lists system users and regular users, each with their username, UID, GID, and home directory/shell. The users listed are: root, daemon, bin, sys, sync, games, man, lp, mail, news, uucp, proxy, www-data, backup, list, irc, gnats, nobody, \_apt, systemd-timesync, systemd-network, systemd-resolve, tss, strongswan, messagebus, usbmux, tcpdump, rtkit, sshd, avahi, nm-openvpn, nm-openconnect, pulse, saned, colord, lightdm, kekalistratova, systemd-coredump, and vboxadd. The user 'kekalistratova' is a regular user with UID 1000, GID 1000, and home directory /home/kekalistratova.

```
root:x:0:0:root:/root:/usr/bin/zsh
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
_apt:x:100:65534::/nonexistent:/usr/sbin/nologin
systemd-timesync:x:101:101:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
systemd-network:x:102:103:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:103:104:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
tss:x:104:110:TPM software stack,,,:/var/lib/tpm:/bin/false
strongswan:x:105:65534::/var/lib/strongswan:/usr/sbin/nologin
messagebus:x:106:111::/nonexistent:/usr/sbin/nologin
usbmux:x:107:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
rtkit:x:109:116:RealtimeKit,,,:/proc:/usr/sbin/nologin
sshd:x:110:65534::/run/sshd:/usr/sbin/nologin
avahi:x:111:118:Avahi mDNS daemon,,,:/run/avahi-daemon:/usr/sbin/nologin
nm-openvpn:x:112:119:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin
nm-openconnect:x:113:120:NetworkManager OpenConnect plugin,,,:/var/lib/NetworkManager:/usr/sbin/nologin
pulse:x:114:121:PulseAudio daemon,,,:/var/run/pulse:/usr/sbin/nologin
saned:x:115:124::/var/lib/saned:/usr/sbin/nologin
colord:x:116:125:colord colour management daemon,,,:/var/lib/colord:/usr/sbin/nologin
lightdm:x:117:126:Light Display Manager:/var/lib/lightdm:/bin/false
kekalistratova:x:1000:1000:kekalistratova,,,:/home/kekalistratova:/usr/bin/zsh
systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
vboxadd:x:998:1::/var/run/vboxadd:/bin/false
```

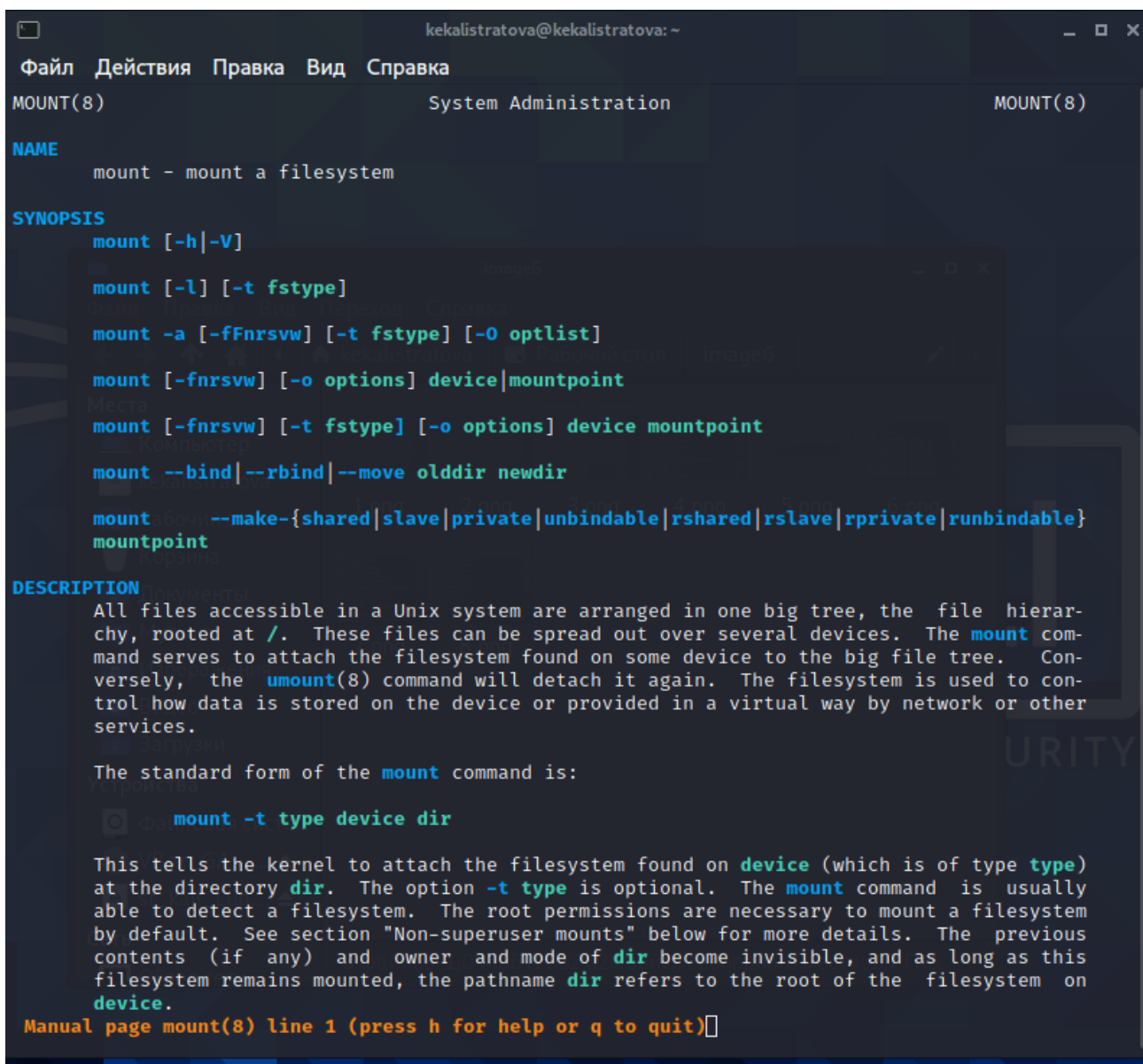
Figure 7: Просмотрим содержимое файла /etc/passwd.



```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
(kekalistratova@kekalistratova)-[~]  
$ mv fun play  
(kekalistratova@kekalistratova)-[~]  
$ cp -r play fun  
(kekalistratova@kekalistratova)-[~]  
$ mv fun play  
(kekalistratova@kekalistratova)-[~]  
$ mv play/fun play/games  
(kekalistratova@kekalistratova)-[~]  
$ chmod u-r feather  
(kekalistratova@kekalistratova)-[~]  
$ cat feather  
cat: feather: Отказано в доступе  
(kekalistratova@kekalistratova)-[~]  
$ cp feather monthly  
cp: невозможно открыть 'feather' для чтения: Отказано в доступе  
(kekalistratova@kekalistratova)-[~]  
$  
(kekalistratova@kekalistratova)-[~]  
$ chmod u+r feather  
(kekalistratova@kekalistratova)-[~]  
$ chmod u-r play  
(kekalistratova@kekalistratova)-[~]  
$ cd play  
(kekalistratova@kekalistratova)-[~/play]  
$ cd ~  
(kekalistratova@kekalistratova)-[~]  
$ chmod u+x play
```

Figure 8: Лишаем владельца файла или каталога прав и даем их ему

- 5) Используя команды «man mount», «man fsck», «man mkfs», «man kill», получим информацию о соответствующих командах.
1. Команда mount: предназначена для монтирования файловой системы. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого дерева обозначается как /. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву. Наиболее часто встречающаяся форма команды mount выглядит следующим образом: «mount -t vfstype device dir» Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типа vfstype, расположенную на устройстве device, к заданному каталогу dir, который часто называют точкой монтирования. (рис. 9)



```
kekalistratova@kekalistratova: ~
Файл Действия Правка Вид Справка
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|slave|rprivate|runbindable}
    mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the file hierar-
    chy, rooted at /. These files can be spread out over several devices. The mount com-
    mand serves to attach the filesystem found on some device to the big file tree. Con-
    versely, the umount(8) command will detach it again. The filesystem is used to con-
    trol 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.

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

Figure 9: Команда man mount

2. Команда fsck: это утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Он использует программы, специфичные для типа файловой системы, которую он проверяет. У команды fsck следующий синтаксис:

`fsck [параметр] -[параметры ФС] [ ... ]`

Например, если нужно восстановить («починить») файловую систему на некотором устройстве /dev/sdb2, следует воспользоваться командой: «`sudo fsck -y /dev/sdb2`». Опция -y необходима, т.к. при её отсутствии придётся слишком часто давать подтверждение. (рис. 10)

```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
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. filesystems  
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 speci-  
fied, 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  
  
The exit status returned when multiple filesystems are checked is the bit-wise OR of  
the exit statuses for each filesystem that is checked.  
  
In actuality, fsck is simply a front-end for the various filesystem checkers  
(fsck.fstype) available under Linux. The filesystem-specific checker is searched for  
in the PATH environment variable. If the PATH is undefined then fallback to /sbin.  
  
Please see the filesystem-specific checker manual pages for further details.  
Manual page fsck(8) line 1 (press h for help or q to quit)
```

Figure 10: Команда `man fsck`

3. Команда `mkfs`: создаёт новую файловую систему Linux. Имеет следующий синтаксис:

`mkfs[-V] [-tfstype] [fs-options] filesystem [blocks]`

`mkfs` используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента `filesystem` для файловой системы может выступать или название устройства (например, `/dev/hda1`, `/dev/sdb2`) или точка монтирования (например, `/`, `usr`, `home`). Аргументом `blocks` указывается количество блоков, которые выделяются для использования этой файловой системой. По окончании работы `mkfs` возвращает 0 - в случае успеха, а 1 - при неудачной операции. Например, команда «`mkfs -t ext2 /dev/hdb1`» создаёт файловую систему типа `ext2` в разделе `/dev/hdb1` (второй жёсткий диск). (рис. 11)

```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
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 ex-  
ecuted. 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 informa-  
tion 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)
```

Figure 11: Команда `man mkfs`

4. Команда `kill`: посылает сигнал процессу или выводит список допустимых сигналов. Имеет следующий синтаксис: `kill [опции] PID`, где PID – это PID (числовой идентификатор) процесса или несколько PID процессов, если требуется послать сигнал сразу нескольким процессам. Например, команда «`kill -KILL 3121`» посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс. (рис. 12)

```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
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. Partic-  
ularly 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.  
  
-l, --list [signal]  
List signal names. This option has optional argument, which will convert sig-  
nal 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.  
  
EXAMPLES  
kill -9 -1  
Kill all processes you can kill.  
  
kill -l 11  
Manual page kill(1) line 1 (press h for help or q to quit)
```

Figure 12: Команда man kill

## Контрольные вопросы

- 1) Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th». (рис. 13)



```
kekalistratova@kekalistratova: ~  
Файл Действия Правка Вид Справка  
(kekalistratova@kekalistratova)-[~]  
$ df -Th  
Файловая система      Тип      Размер  Использовано  Дост  Использовано%  Смонтирован  
о в  
udev                  devtmpfs  2,4G      0      2,4G      0% /dev  
tmpfs                 tmpfs     494M     944K     493M      1% /run  
/dev/mapper/kekalistratova--vg-root ext4      6,4G     3,4G     2,8G     56% /  
tmpfs                 tmpfs     2,5G     154M     2,3G      7% /dev/shm  
tmpfs                 tmpfs     5,0M      0      5,0M      0% /run/lock  
/dev/sda1             ext2      470M      60M     386M     14% /boot  
kali_lunix_           vboxsf    238G     208G     30G     88% /media/sf_k  
ali_lunix_  
tmpfs                 tmpfs     494M      64K     494M      1% /run/user/1  
000
```

Figure 13: Команда `df -Th`

Из рисунка видно, что на моем компьютере есть следующие файловые системы: `devtmpfs`, `tmpfs`, `ext4`.

`devtmpfs` 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 «`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.

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

Так же 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 псевдо файловая 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 кеши, 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 –съёмные 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) 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
- 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 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, 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-r –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 способзаписи:

= установить право

- лишить права
- датьправог чтение

w запись

x выполнение

u (user) владелец файла

g (group) группа, к которой принадлежит владелец файла

o (others) все остальные

## Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов, получила навыкипо применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.