

Отчёта по лабораторной работе №5:

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

Слуцкая Евгения Александровна

Содержание

1	Цель работы	4
2	Задание	5
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Контрольные вопросы	24
6	Выводы	32
	Список литературы	33

Список иллюстраций

4.1	Копирование файла в текущем каталоге	9
4.2	Копирование нескольких файлов в каталог	10
4.3	Копирование файлов в произвольном каталоге	10
4.4	Копирование каталогов в текущем и произвольном каталогах . .	10
4.5	Переименование и перемещение файлов в другой каталог	11
4.6	Переименование и перемещение каталога	11
4.7	Установка прав	12
4.8	Копирование файла io.h	12
4.9	Создание директории, перемещение и переименование файла . .	12
4.10	Создание файла abc1 и копирование его в каталог ski.places . . .	13
4.11	Перемещение файлов в новый каталог	13
4.12	Новый каталог	13
4.13	Изменение прав каталога australia	14
4.14	Изменение прав каталога play	15
4.15	Изменение прав файла my_os	15
4.16	Изменение прав файла feathers	16
4.17	Содержимое файла /etc/passwd	17
4.18	Копирование и перемещение файла	18
4.19	Перемещение каталога	18
4.20	Изменение прав владельца файла	19
4.21	Изменение прав владельца каталога	19
4.22	Команда mount	20
4.23	Команда fsck	21
4.24	Команда mkfs	22
4.25	Команда kill	23

1 Цель работы

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

2 Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
 - Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.
 - В домашнем каталоге создайте директорию `~/ski.places`.
 - Переместите файл `equipment` в каталог `~/ski.places`.
 - Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 - Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 - Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 - Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 - Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

- drwxr-r- ... australia
- drwx-x-x ... play
- -r-xr-r- ... my_os
- -rw-rw-r- ... feathers

При необходимости создайте нужные файлы.

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

- Просмотрите содержимое файла /etc/password.
- Скопируйте файл ~/feathers в файл ~/file.old.
- Переместите файл ~/file.old в каталог ~/play.
- Скопируйте каталог ~/play в каталог ~/fun.
- Переместите каталог ~/fun в каталог ~/play и назовите его games.
- Лишите владельца файла ~/feathers права на чтение.
- Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?
- Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?
- Дайте владельцу файла ~/feathers право на чтение.
- Лишите владельца каталога ~/play права на выполнение.
- Перейдите в каталог ~/play. Что произошло?
- Дайте владельцу каталога ~/play право на выполнение.

5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

3 Теоретическое введение

Приведём основные команды Unix, которые мы будем использовать в процессе выполнения лабораторной работы №4.

В табл. tbl. 3.1 приведен краткий справочник по командам терминала Unix-систем.

Таблица 3.1: Краткий справочник по командам терминала GNU Linux

Имя ко- манды	Описание команды
touch имя_файла	Создание файла
cat имя_файла	Просмотр содержимого файлов небольшого размера
less имя_файла	Просмотр файла постранично
head [-n] имя_файла	Вывод по умолчанию 10 первых строк файла или n строк
tail [-n] имя_файла	Вывод последних n строк файла

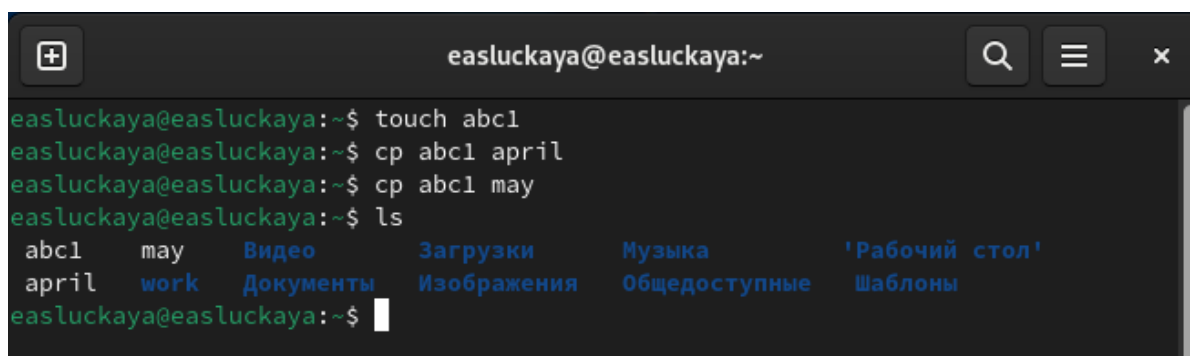
Имя ко- манды	Описание команды
cp	Копирование файлов и каталогов
исход- ная_директория	
целе- вая_директория	
mkdir	Создание директории
имя_директории	
mv	Перемещение (переименование) каталогов
ста- рый_файл	
но- вый_файл	

4 Выполнение лабораторной работы

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

Для копирования файлов и каталогов будем использовать команду `cp [-опции] исходный_файл целевой_файл`, а для их перемещения и переименования будем использовать команду `mv [-опции] старый_файл новый_файл`:

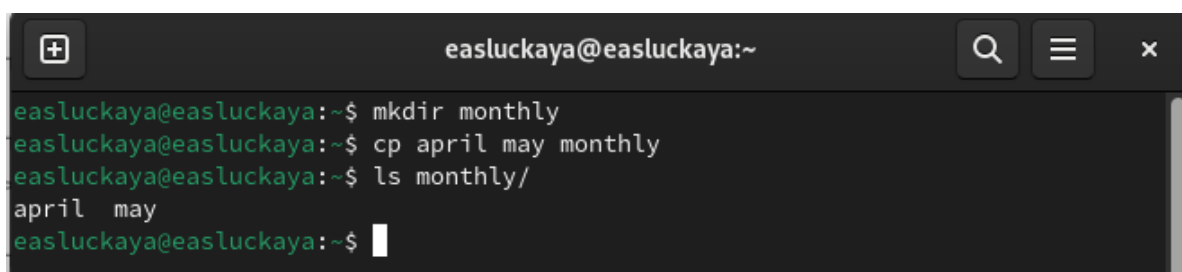
Копирование файла в текущем каталоге - скопировать файл `~/abc1` в файл `april` и в файл `may` (рис. 4.1).

A screenshot of a terminal window titled 'easluckaya@easluckaya:~'. The terminal shows the following commands and output:

```
easluckaya@easluckaya:~$ touch abc1
easluckaya@easluckaya:~$ cp abc1 april
easluckaya@easluckaya:~$ cp abc1 may
easluckaya@easluckaya:~$ ls
abc1    may    Видео    Загрузки    Музыка    'Рабочий стол'
april   work   Документы  Изображения  Общедоступные  Шаблоны
easluckaya@easluckaya:~$
```

Рис. 4.1: Копирование файла в текущем каталоге

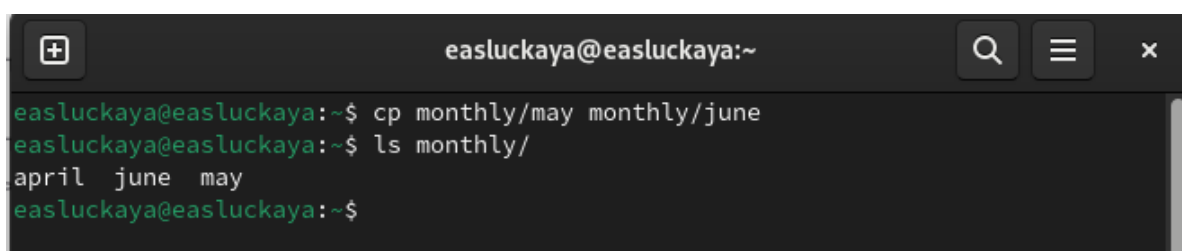
Копирование нескольких файлов в каталог - скопировать файлы `april` и `may` в каталог `monthly` (рис. 4.2).



```
easluckaya@easluckaya:~$ mkdir monthly
easluckaya@easluckaya:~$ cp april may monthly
easluckaya@easluckaya:~$ ls monthly/
april  may
easluckaya@easluckaya:~$
```

Рис. 4.2: Копирование нескольких файлов в каталог

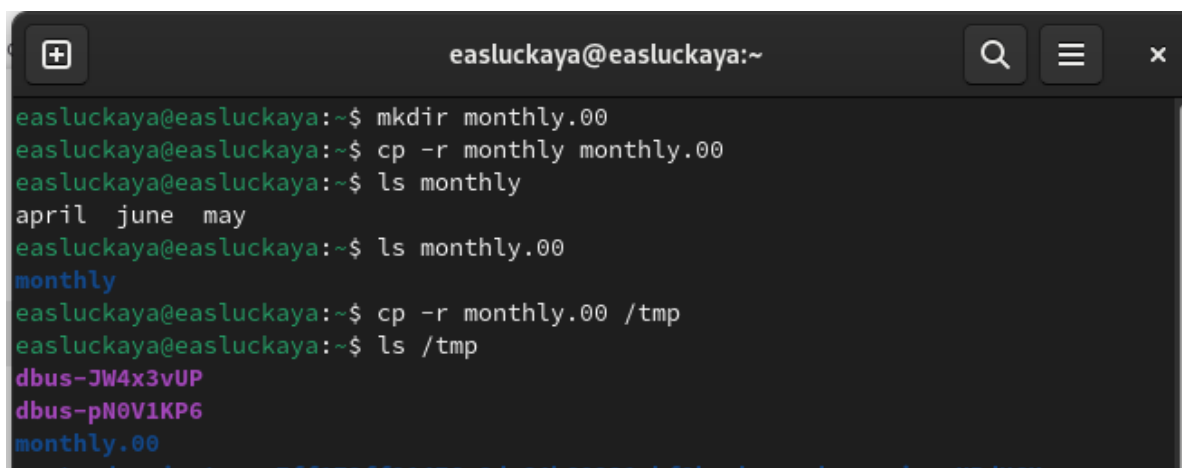
Копирование файлов в произвольном каталоге - скопировать файл `monthly/may` в файл с именем `june` (рис. 4.3).



```
easluckaya@easluckaya:~$ cp monthly/may monthly/june
easluckaya@easluckaya:~$ ls monthly/
april  june  may
easluckaya@easluckaya:~$
```

Рис. 4.3: Копирование файлов в произвольном каталоге

Копирование каталогов в текущем и произвольном каталогах - скопировать каталог `monthly` в каталог `monthly.00`, скопировать каталог `monthly.00` в каталог `/tmp` (рис. 4.4).

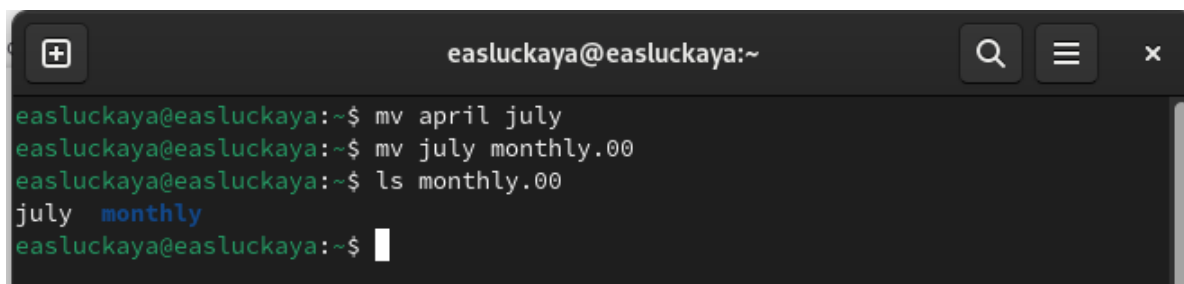


```
easluckaya@easluckaya:~$ mkdir monthly.00
easluckaya@easluckaya:~$ cp -r monthly monthly.00
easluckaya@easluckaya:~$ ls monthly
april  june  may
easluckaya@easluckaya:~$ ls monthly.00
monthly
easluckaya@easluckaya:~$ cp -r monthly.00 /tmp
easluckaya@easluckaya:~$ ls /tmp
dbus-JW4x3vUP
dbus-pN0V1KP6
monthly.00
systemd-private-7551725691676-94dc4b69231ebf0b-bb9ed-222000-184000
```

Рис. 4.4: Копирование каталогов в текущем и произвольном каталогах

Переименование файлов в текущем каталоге - изменить название файла

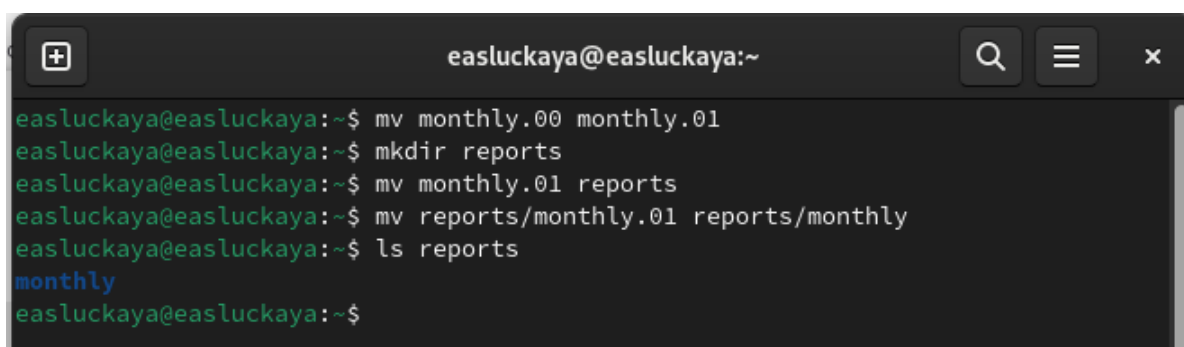
april на july в домашнем каталоге. **Перемещение файлов в другой каталог** - переместить файл july в каталог monthly.00(рис. 4.5).

A terminal window titled 'easluckaya@easluckaya:~' with search, menu, and close buttons. It shows the following commands and output:

```
easluckaya@easluckaya:~$ mv april july
easluckaya@easluckaya:~$ mv july monthly.00
easluckaya@easluckaya:~$ ls monthly.00
july  monthly
easluckaya@easluckaya:~$
```

Рис. 4.5: Переименование и перемещение файлов в другой каталог

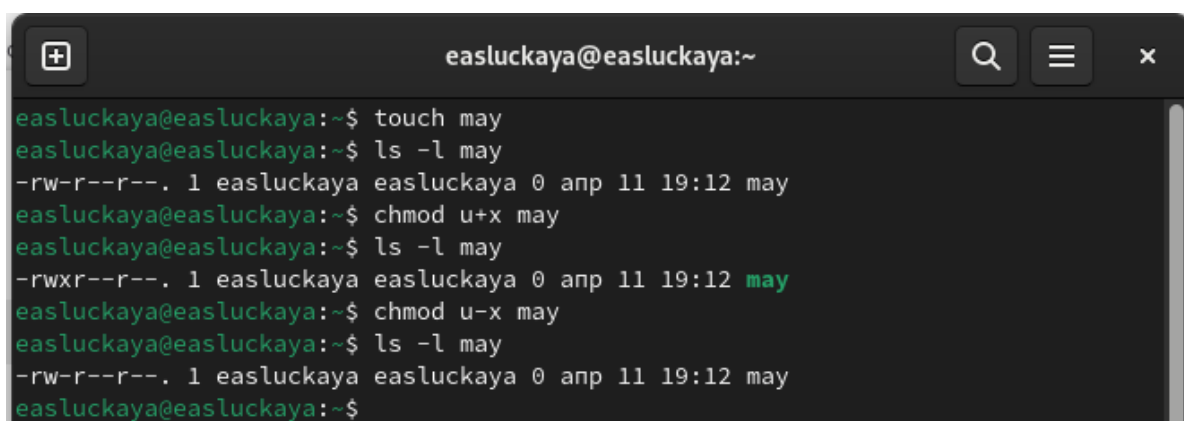
Переименование каталогов в текущем каталоге - переименовать каталог monthly.00 в monthly.01. **Перемещение каталога в другой каталог** - переместить каталог monthly.01 в каталог reports. **Переименование каталога, не являющегося текущим** - переименовать каталог reports/monthly.01 в reports/monthly (рис. 4.6).

A terminal window titled 'easluckaya@easluckaya:~' with search, menu, and close buttons. It shows the following commands and output:

```
easluckaya@easluckaya:~$ mv monthly.00 monthly.01
easluckaya@easluckaya:~$ mkdir reports
easluckaya@easluckaya:~$ mv monthly.01 reports
easluckaya@easluckaya:~$ mv reports/monthly.01 reports/monthly
easluckaya@easluckaya:~$ ls reports
monthly
easluckaya@easluckaya:~$
```

Рис. 4.6: Переименование и перемещение каталога

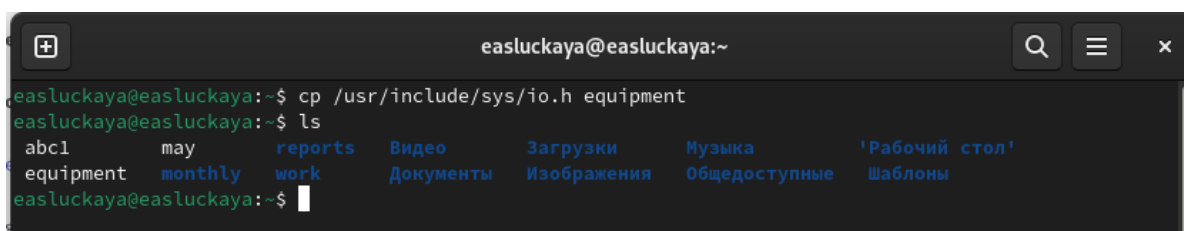
Создадим файл ~/may с правом выполнения для владельца, а затем лишим владельца прав на выполнение (рис. 4.7).



```
easluckaya@easluckaya:~$ touch may
easluckaya@easluckaya:~$ ls -l may
-rw-r--r--. 1 easluckaya easluckaya 0 апр 11 19:12 may
easluckaya@easluckaya:~$ chmod u+x may
easluckaya@easluckaya:~$ ls -l may
-rwxr--r--. 1 easluckaya easluckaya 0 апр 11 19:12 may
easluckaya@easluckaya:~$ chmod u-x may
easluckaya@easluckaya:~$ ls -l may
-rw-r--r--. 1 easluckaya easluckaya 0 апр 11 19:12 may
easluckaya@easluckaya:~$
```

Рис. 4.7: Установка прав

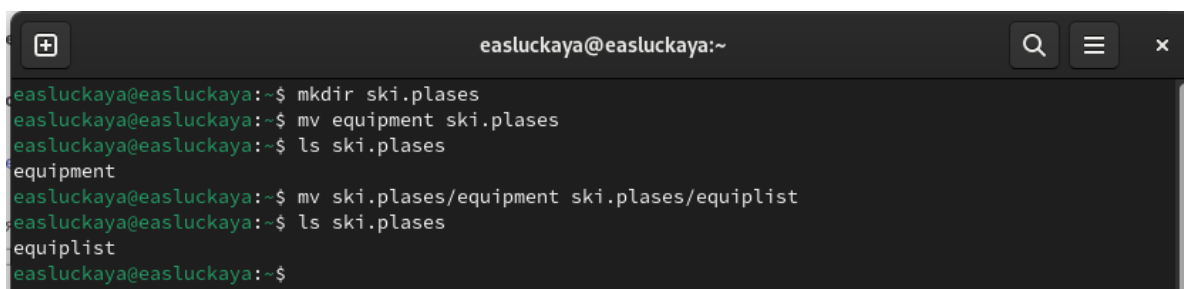
2. Скопируем файл `/usr/include/sys/io.h` в домашний каталог и назовем его `equipment`. Если файла `io.h` нет, то используем любой другой файл в каталоге `/usr/include/sys/` вместо него (рис. 4.8).



```
easluckaya@easluckaya:~$ cp /usr/include/sys/io.h equipment
easluckaya@easluckaya:~$ ls
abcl1    may      reports  Видео    Загрузки  Музыка    'Рабочий стол'
equipment monthly  work     Документы  Изображения  Общедоступные  Шаблоны
easluckaya@easluckaya:~$
```

Рис. 4.8: Копирование файла `io.h`

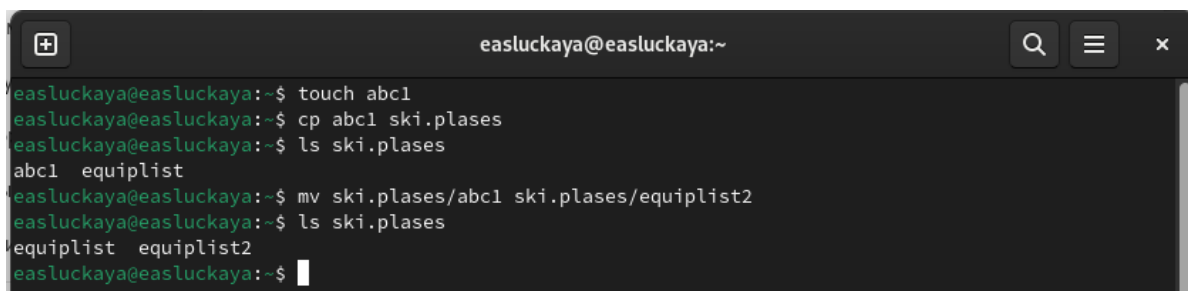
В домашнем каталоге создадим директорию `~/ski.places` и переместим файл `equipment` в новый каталог. После переименуем файл `equiplist` (рис. 4.9).



```
easluckaya@easluckaya:~$ mkdir ski.places
easluckaya@easluckaya:~$ mv equipment ski.places
easluckaya@easluckaya:~$ ls ski.places
equipment
easluckaya@easluckaya:~$ mv ski.places/equipment ski.places/equiplist
easluckaya@easluckaya:~$ ls ski.places
equiplist
easluckaya@easluckaya:~$
```

Рис. 4.9: Создание директории, перемещение и переименование файла

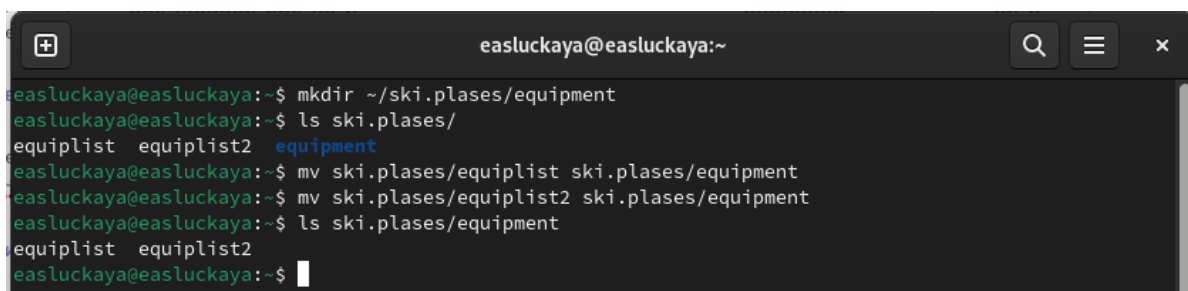
Создадим в домашнем каталоге файл `abc1` и скопируем его в каталог `~/ski.plases` и назовем его `equiplist2` (рис. 4.10).



```
easluckaya@easluckaya:~$ touch abc1
easluckaya@easluckaya:~$ cp abc1 ski.plases
easluckaya@easluckaya:~$ ls ski.plases
abc1  equiplist
easluckaya@easluckaya:~$ mv ski.plases/abc1 ski.plases/equiplist2
easluckaya@easluckaya:~$ ls ski.plases
equiplist  equiplist2
easluckaya@easluckaya:~$
```

Рис. 4.10: Создание файла `abc1` и копирование его в каталог `ski.plases`

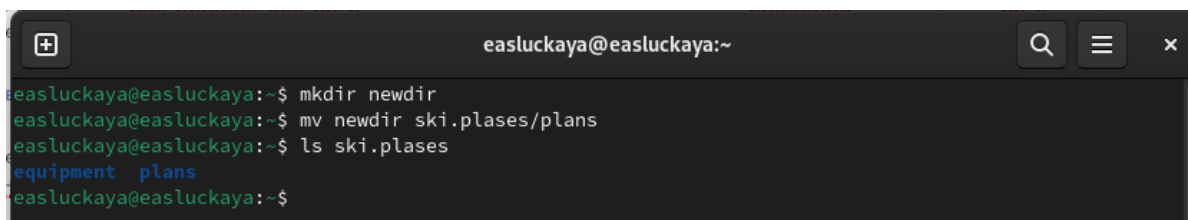
Создадим каталог с именем `equipment` в каталоге `~/ski.plases`. Затем переместим файлы `~/ski.plases/equiplist` и `equiplist2` в каталог `~/ski.plases/equipment` (рис. 4.11).



```
easluckaya@easluckaya:~$ mkdir ~/ski.plases/equipment
easluckaya@easluckaya:~$ ls ski.plases/
equiplist  equiplist2  equipment
easluckaya@easluckaya:~$ mv ski.plases/equiplist ski.plases/equipment
easluckaya@easluckaya:~$ mv ski.plases/equiplist2 ski.plases/equipment
easluckaya@easluckaya:~$ ls ski.plases/equipment
equiplist  equiplist2
easluckaya@easluckaya:~$
```

Рис. 4.11: Перемещение файлов в новый каталог

Создадим и переместим каталог `~/newdir` в каталог `~/ski.plases` и назовем его `plans`. (рис. 4.12).



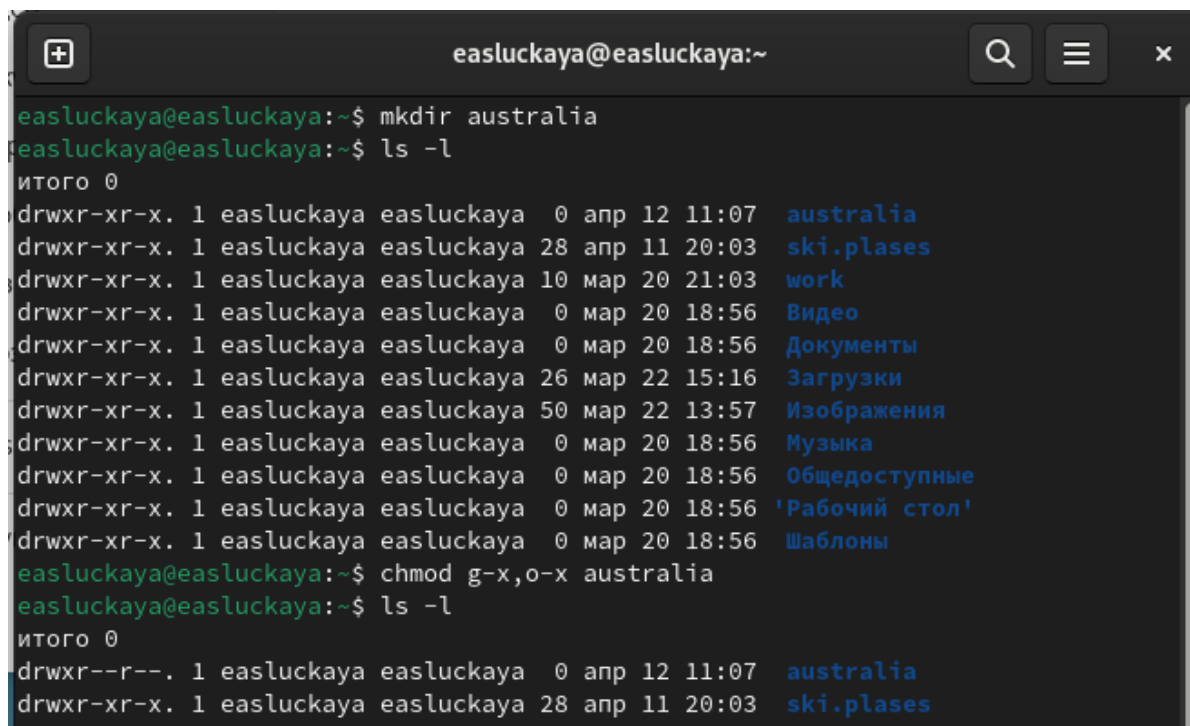
```
easluckaya@easluckaya:~$ mkdir newdir
easluckaya@easluckaya:~$ mv newdir ski.plases/plans
easluckaya@easluckaya:~$ ls ski.plases
equipment  plans
easluckaya@easluckaya:~$
```

Рис. 4.12: Новый каталог

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

в начале таких прав нет:

- **drwxr-r- ... australia:** создадим новый каталог `australia`, проверим какие у него права, а после, в моем случае, используем `chmod g-x,o-x australia`, чтобы изменить на нужные права (рис. 4.13).



```
easluckaya@easluckaya:~$ mkdir australia
easluckaya@easluckaya:~$ ls -l
итого 0
drwxr-xr-x. 1 easluckaya easluckaya  0 anp 12 11:07 australia
drwxr-xr-x. 1 easluckaya easluckaya 28 anp 11 20:03 ski.plases
drwxr-xr-x. 1 easluckaya easluckaya 10 map 20 21:03 work
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 Видео
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 Документы
drwxr-xr-x. 1 easluckaya easluckaya 26 map 22 15:16 Загрузки
drwxr-xr-x. 1 easluckaya easluckaya 50 map 22 13:57 Изображения
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 Музыка
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 Общедоступные
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 'Рабочий стол'
drwxr-xr-x. 1 easluckaya easluckaya  0 map 20 18:56 Шаблоны
easluckaya@easluckaya:~$ chmod g-x,o-x australia
easluckaya@easluckaya:~$ ls -l
итого 0
drwxr--r--. 1 easluckaya easluckaya  0 anp 12 11:07 australia
drwxr-xr-x. 1 easluckaya easluckaya 28 anp 11 20:03 ski.plases
```

Рис. 4.13: Изменение прав каталога `australia`

- **drwx-x-x ... play:** делаем то же самое, но используем `chmod g-r,o-r play` (рис. 4.14).

```
easluckaya@easluckaya:~$ mkdir play
easluckaya@easluckaya:~$ ls -l
итого 0
drwxr--r--. 1 easluckaya easluckaya  0 апр 12 11:07 australia
drwxr-xr-x. 1 easluckaya easluckaya  0 апр 12 11:09 play
drwxr-xr-x. 1 easluckaya easluckaya 28 апр 11 20:03 ski.places
drwxr-xr-x. 1 easluckaya easluckaya 10 мар 20 21:03 work
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 Видео
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 Документы
drwxr-xr-x. 1 easluckaya easluckaya 26 мар 22 15:16 Загрузки
drwxr-xr-x. 1 easluckaya easluckaya 50 мар 22 13:57 Изображения
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 Музыка
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 Общедоступные
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 'Рабочий стол'
drwxr-xr-x. 1 easluckaya easluckaya  0 мар 20 18:56 Шаблоны
easluckaya@easluckaya:~$ chmod g-r,o-r play
easluckaya@easluckaya:~$ ls -l
итого 0
drwxr--r--. 1 easluckaya easluckaya  0 апр 12 11:07 australia
drwx--x--x. 1 easluckaya easluckaya  0 апр 12 11:09 play
drwxr-xr-x. 1 easluckaya easluckaya 28 апр 11 20:03 ski.places
```

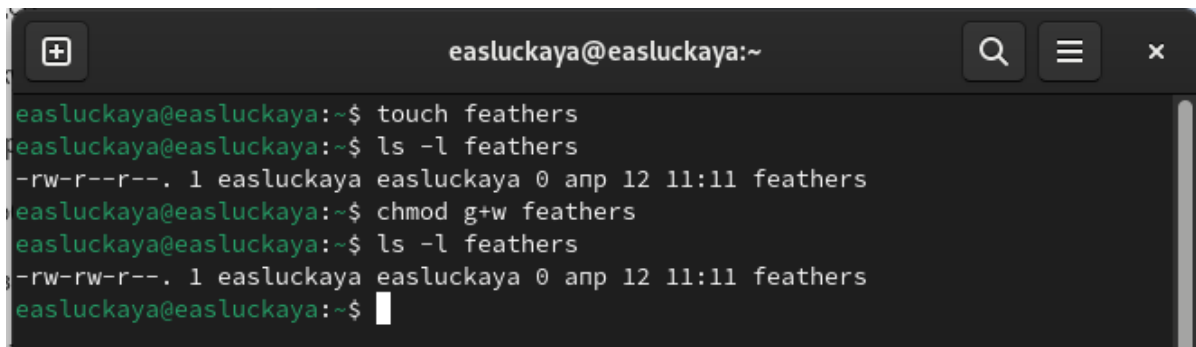
Рис. 4.14: Изменение прав каталога play

- **-r-xr-r-** ... **my_os**: создаем в этот раз уже файл my_os и пишем команду `chmod u-w,u+x my-os` (рис. 4.15).

```
easluckaya@easluckaya:~$ touch my_os
easluckaya@easluckaya:~$ ls -l my_os
-rw-r--r--. 1 easluckaya easluckaya 0 апр 12 11:10 my_os
easluckaya@easluckaya:~$ chmod u-w,u+x my_os
easluckaya@easluckaya:~$ ls -l my_os
-r-xr--r--. 1 easluckaya easluckaya 0 апр 12 11:10 my_os
easluckaya@easluckaya:~$
```

Рис. 4.15: Изменение прав файла my_os

- **-rw-rw-r-** ... **feathers**: здесь уже используем `chmod g+w feathers` (рис. 4.16).

A terminal window titled 'easluckaya@easluckaya:~' with search, menu, and close buttons. It shows a sequence of commands: 'touch feathers', 'ls -l feathers' (output: '-rw-r--r--. 1 easluckaya easluckaya 0 anp 12 11:11 feathers'), 'chmod g+w feathers', and 'ls -l feathers' (output: '-rw-rw-r--. 1 easluckaya easluckaya 0 anp 12 11:11 feathers').

```
easluckaya@easluckaya:~$ touch feathers
easluckaya@easluckaya:~$ ls -l feathers
-rw-r--r--. 1 easluckaya easluckaya 0 anp 12 11:11 feathers
easluckaya@easluckaya:~$ chmod g+w feathers
easluckaya@easluckaya:~$ ls -l feathers
-rw-rw-r--. 1 easluckaya easluckaya 0 anp 12 11:11 feathers
easluckaya@easluckaya:~$
```

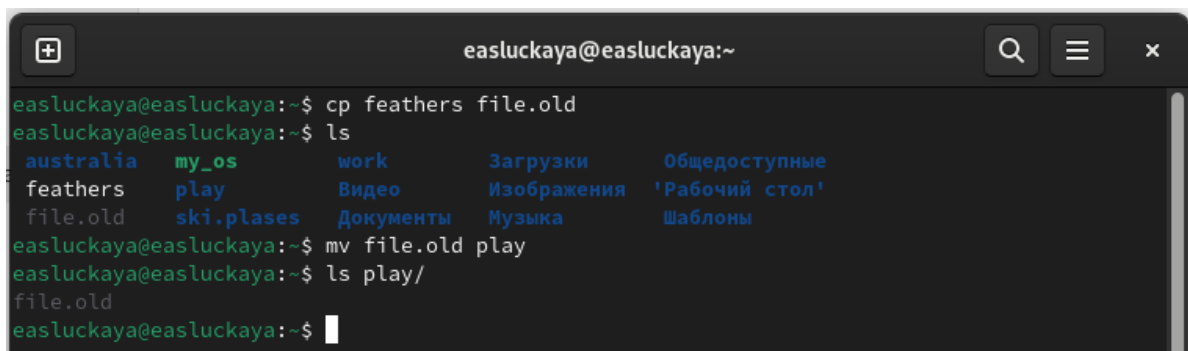
Рис. 4.16: Изменение прав файла feathers

4. Просмотрим содержимое файла `/etc/passwd`, используя команду **cat** (рис. 4.17).


```
easluckaya@easluckaya:~$ cat /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
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:/var/spool/mail:/usr/sbin/nologin
operator:x:11:0:operator:/root:/usr/sbin/nologin
games:x:12:100:games:/usr/games:/usr/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/usr/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/usr/sbin/nologin
dbus:x:81:81:System Message Bus:/usr/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
tss:x:59:59:Account used for TPM access:/usr/sbin/nologin
systemd-coredump:x:998:998:systemd Core Dumper:/usr/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/usr/sbin/nologin
systemd-oom:x:997:997:systemd Userspace OOM Killer:/usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/usr/sbin/nologin
systemd-timesync:x:996:996:systemd Time Synchronization:/usr/sbin/nologin
qemu:x:107:107:qemu user:/sbin/nologin
polkitd:x:114:114>User for polkitd:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
geoclue:x:995:994>User for geoclue:/var/lib/geoclue:/sbin/nologin
nm-openconnect:x:994:993:NetworkManager user for OpenConnect:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/sbin/nologin
gluster:x:993:992:GlusterFS daemons:/run/gluster:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/proc:/sbin/nologin
pipewire:x:992:990:PipeWire System Daemon:/run/pipewire:/usr/sbin/nologin
saslauthd:x:991:76:Saslauthd user:/run/saslauthd:/sbin/nologin
chrony:x:990:989:chrony system user:/var/lib/chrony:/sbin/nologin
dnsmasq:x:989:988:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/usr/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
openvpn:x:988:987:OpenVPN:/etc/openvpn:/sbin/nologin
nm-openvpn:x:987:986:Default user for running openvpn spawned by NetworkManager:/sbin/nologin
colord:x:986:985>User for colord:/var/lib/colord:/sbin/nologin
unbound:x:985:984:Unbound DNS resolver:/var/lib/unbound:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
flatpak:x:984:982:Flatpak system helper:/usr/sbin/nologin
gdm:x:42:42:GNOME Display Manager:/var/lib/gdm:/usr/sbin/nologin
gnome-initial-setup:x:983:981:/run/gnome-initial-setup:/sbin/nologin
vboxadd:x:982:1:/var/run/vboxadd:/sbin/nologin
sshd:x:74:74:Privilege-separated SSH:/usr/share/empty.sshd:/usr/sbin/nologin
tcpdump:x:72:72:tcpdump:/usr/sbin/nologin
easluckaya:x:1000:1000:easluckaya:/home/easluckaya:/bin/bash
easluckaya@easluckaya:~$
```

Рис. 4.17: Содержимое файла /etc/passwd

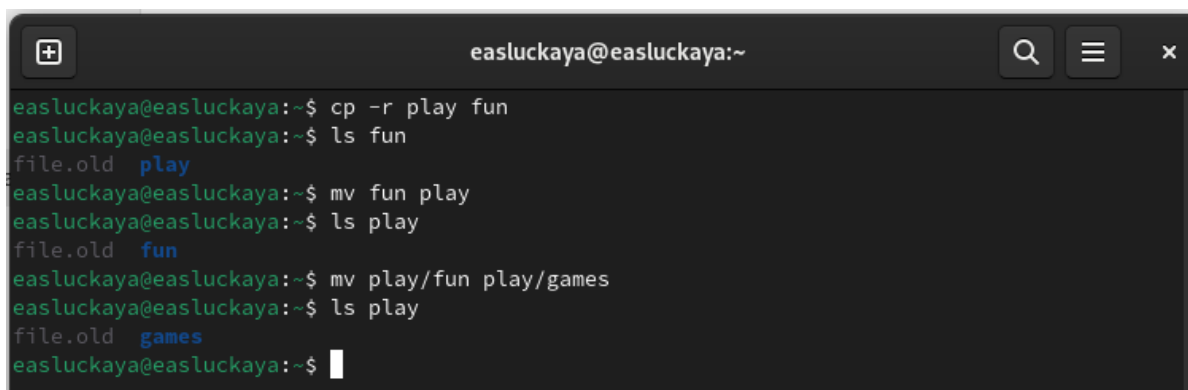
Скопируем файл ~/feathers в файл ~/file.old, а затем переместим файл ~/file.old в каталог ~/play (рис. 4.18).

A terminal window titled 'easluckaya@easluckaya:~' with search, menu, and close buttons. The command history shows: 'cp feathers file.old', 'ls' (listing files like australia, my_os, work, Загрузки, Общедоступные, feathers, play, Видео, Изображения, 'Рабочий стол', file.old, ski.plases, Документы, Музыка, Шаблоны), 'mv file.old play', and 'ls play/' (listing file.old).

```
easluckaya@easluckaya:~$ cp feathers file.old
easluckaya@easluckaya:~$ ls
australia  my_os      work      Загрузки  Общедоступные
feathers    play      Видео     Изображения  'Рабочий стол'
file.old   ski.plases  Документы  Музыка      Шаблоны
easluckaya@easluckaya:~$ mv file.old play
easluckaya@easluckaya:~$ ls play/
file.old
easluckaya@easluckaya:~$
```

Рис. 4.18: Копирование и перемещение файла

Скопируем каталог ~/play в каталог ~/fun. После переместим каталог ~/fun в каталог ~/play и назовите его games(рис. 4.19).

A terminal window titled 'easluckaya@easluckaya:~' with search, menu, and close buttons. The command history shows: 'cp -r play fun', 'ls fun' (listing file.old, play), 'mv fun play', 'ls play' (listing file.old, fun), 'mv play/fun play/games', 'ls play' (listing file.old, games), and the prompt.

```
easluckaya@easluckaya:~$ cp -r play fun
easluckaya@easluckaya:~$ ls fun
file.old  play
easluckaya@easluckaya:~$ mv fun play
easluckaya@easluckaya:~$ ls play
file.old  fun
easluckaya@easluckaya:~$ mv play/fun play/games
easluckaya@easluckaya:~$ ls play
file.old  games
easluckaya@easluckaya:~$
```

Рис. 4.19: Перемещение каталога

Лишим владельца файла ~/feathers права на чтение, используя команду `chmod u-r feathers`. Если мы попробуем просмотреть файл ~/feathers командой `cat`, то нам будет отказано в доступе, то же самое произойдет, если захотим скопировать данный файл. После вернем все обратно командой `chmod u+r feathers` (рис. 4.20).

```
easluckaya@easluckaya:~  
easluckaya@easluckaya:~$ ls -l feathers  
-rw-rw-r--. 1 easluckaya easluckaya 0 апр 12 11:11 feathers  
easluckaya@easluckaya:~$ chmod u-r feathers  
easluckaya@easluckaya:~$ ls -l feathers  
--w-rw-r--. 1 easluckaya easluckaya 0 апр 12 11:11 feathers  
easluckaya@easluckaya:~$ cat feathers  
cat: feathers: Отказано в доступе  
easluckaya@easluckaya:~$ cp feathers play  
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе  
easluckaya@easluckaya:~$ chmod u+r feathers  
easluckaya@easluckaya:~$ ls -l feathers  
-rw-rw-r--. 1 easluckaya easluckaya 0 апр 12 11:11 feathers  
easluckaya@easluckaya:~$
```

Рис. 4.20: Изменение прав владельца файла

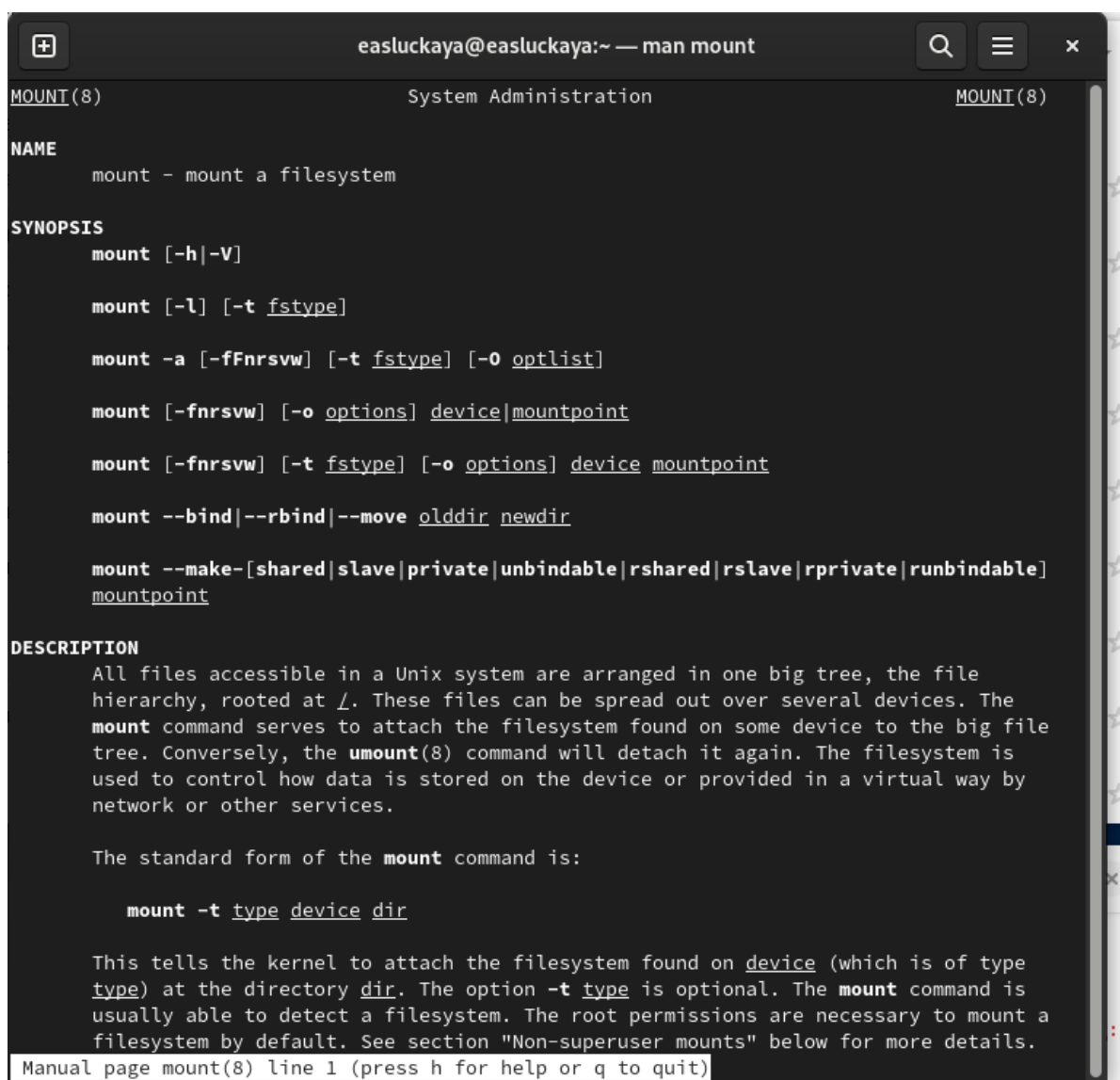
Лишим теперь владельца каталога ~/play права на выполнение, командой `chmod u-x play`. Попробуем перейти в каталог ~/play, то нам будет отказано в доступе. Вернем владельцу каталога ~/play право на выполнение, командой `chmod u+x play` (рис. 4.21).

```
easluckaya@easluckaya:~  
easluckaya@easluckaya:~$ chmod u-x play  
easluckaya@easluckaya:~$ ls -l play  
ls: невозможно получить доступ к 'play/file.old': Отказано в доступе  
ls: невозможно получить доступ к 'play/games': Отказано в доступе  
итого 0  
-????????? ? ? ? ? ? file.old  
d????????? ? ? ? ? ? games  
easluckaya@easluckaya:~$ cd play  
bash: cd: play: Отказано в доступе  
easluckaya@easluckaya:~$ chmod u+x play  
easluckaya@easluckaya:~$ ls -l play  
итого 0  
-rw-r--r--. 1 easluckaya easluckaya 0 апр 12 11:36 file.old  
drwx--x--x. 1 easluckaya easluckaya 24 апр 12 11:44 games  
easluckaya@easluckaya:~$
```

Рис. 4.21: Изменение прав владельца каталога

Прочитаем man по командам `mount`, `fsck`, `mkfs`, `kill`:

- **mount**: команда `mount` используется для подключения файловых систем к каталогу в иерархии файловой системы (рис. 4.22).



```
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:

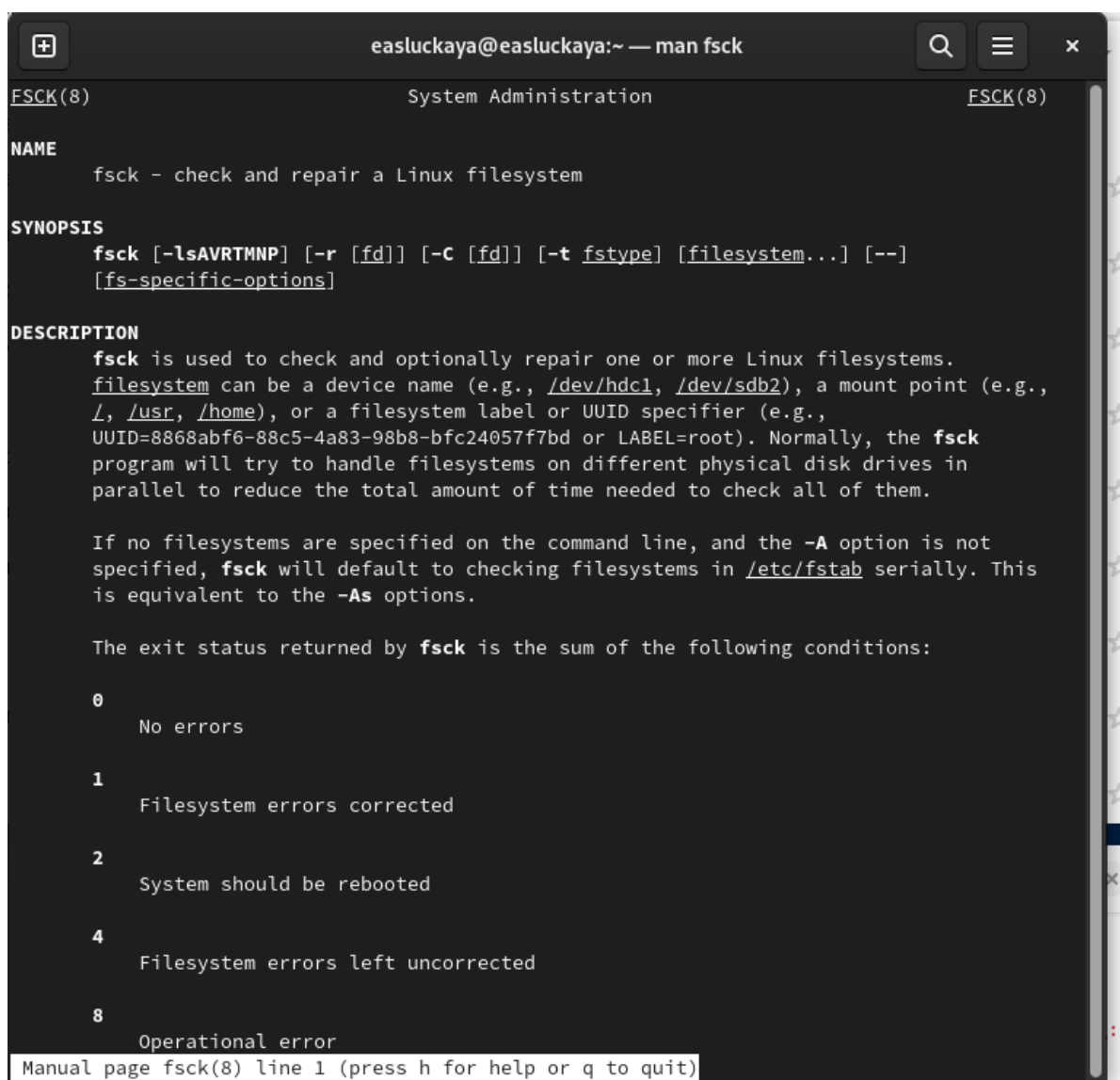
        mount -t type device dir

    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.

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

Рис. 4.22: Команда mount

- **fsck**: команда fsck (file system check) предназначена для проверки и исправления целостности файловых систем (рис. 4.23).



```
easluckaya@easluckaya:~ — man fsck
FSCK(8)                                System Administration                                FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

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

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

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

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

    0      No errors

    1      Filesystem errors corrected

    2      System should be rebooted

    4      Filesystem errors left uncorrected

    8      Operational error

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

Рис. 4.23: Команда fsck

- **mkfs**: команда mkfs (make file system) используется для создания новой файловой системы на указанном устройстве (рис. 4.24).

```
easluckaya@easluckaya:~ — man mkfs
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.

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

Рис. 4.24: Команда mkfs

- **kill**: команда kill используется для отправки сигнала процессу или группе процессов, что может привести к их завершению (рис. 4.25).

```
easluckaya@easluckaya:~ — man kill
KILL(1) User Commands KILL(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--]
    pid|name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified processes or process
    groups.

    If no signal is specified, the TERM signal is sent. The default action for this
    signal is to terminate the process. This signal should be used in preference to
    the KILL signal (number 9), since a process may install a handler for the TERM
    signal in order to perform clean-up steps before terminating in an orderly
    fashion. If a process does not terminate after a TERM signal has been sent, then
    the KILL signal may be used; be aware that the latter signal cannot be caught, and
    so does not give the target process the opportunity to perform any clean-up before
    terminating.

    Most modern shells have a builtin kill command, with a usage rather similar to
    that of the command described here. The --all, --pid, and --queue options, and the
    possibility to specify processes by command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking is still
    performed.

ARGUMENTS
    The list of processes to be signaled can be a mixture of names and PIDs.

    pid
        Each pid can be expressed in one of the following ways:
```

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

Рис. 4.25: Команда kill

5 Контрольные вопросы

1. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.

Общая структура файловой системы в большинстве операционных систем может быть представлена следующим образом:

- / - корневая директория
- /bin** - содержит основные исполняемые файлы, необходимые для загрузки и базовых операций в системе.
- /boot** - содержит файлы, необходимые для загрузки операционной системы.
- /dev** - содержит файлы, представляющие устройства в системе.
- /etc** - содержит конфигурационные файлы для системы и установленных программ.
- /home** - директории пользователей.
- /lib** - содержит библиотеки, необходимые для запуска исполняемых файлов в **/bin** и **/sbin**.
- /media** - место для монтирования временных носителей, таких как USB-накопители.
- /mnt** - место для временного монтирования файловых систем.
- /opt** - используется для установки “дополнительных” программ.
- /proc** - виртуальная файловая система, предоставляющая информацию о процессах в системе.
- /root** - домашняя директория суперпользователя (root).

/run - временная файловая система, хранящая данные, специфичные для текущего запуска системы.

/sbin - содержит основные системные исполняемые файлы.

/srv - содержит данные, используемые локальными сервисами.

/sys - виртуальная файловая система, предоставляющая информацию о ядре Linux.

/tmp - временная директория, доступная всем пользователям для временных файлов.

/usr - вторичная иерархия, содержащая все несистемные программы и данные.

/var - содержит изменяемые файлы, такие как журналы, временные файлы и кэши.

Каждая из этих директорий первого уровня имеет свою уникальную роль в файловой системе операционной системы.

2. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?

Чтобы содержимое некоторой файловой системы стало доступным операционной системе, необходимо выполнить операцию монтирования (**mount**). Монтирование - это процесс подключения файловой системы к определенной точке в иерархии файловой системы операционной системы. Это позволяет операционной системе видеть и обращаться к содержимому этой файловой системы.

3. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Основные причины нарушения целостности файловой системы включают:

- **Сбои в питании:** Внезапное отключение питания может привести к повреждению файлов, так как операции записи могут быть прерваны, что приводит к неправильному состоянию файловой системы.

- **Сбои в работе жёсткого диска или других устройств хранения данных:** Неисправности жестких дисков или других устройств хранения данных могут вызвать повреждение файлов и структуры файловой системы.
- **Вирусы и вредоносное ПО:** Атаки вирусов и вредоносного программного обеспечения могут изменять или удалять файлы, что приводит к нарушению целостности файловой системы.
- **Неправильное завершение работы программ или операционной системы:** Если программа или операционная система завершается некорректно, это может привести к повреждению файлов и структуры файловой системы.
- **Физические повреждения устройств хранения данных:** Механические повреждения жестких дисков или других устройств хранения данных могут привести к потере данных и повреждению файловой системы.

Для устранения повреждений файловой системы можно предпринять следующие шаги:

- **Проверка и восстановление файловой системы:** Использование специальных инструментов, таких как fsck в Linux, для проверки и восстановления целостности файловой системы.
- **Использование резервных копий:** Восстановление поврежденных файлов из резервных копий, если они доступны.
- **Использование специализированных программ:** Использование специализированных программ для восстановления данных с поврежденных устройств хранения.
- **Замена неисправных устройств хранения данных:** В случае физических повреждений устройств хранения данных требуется их замена и восстановление данных из резервных копий, если таковые имеются.

- **Установка антивирусного ПО и обновление системы:** Защита от вирусов и вредоносного ПО путем установки антивирусного программного обеспечения и регулярное обновление операционной системы для устранения уязвимостей.

Кроме того, важно регулярно создавать резервные копии данных, чтобы в случае повреждения файловой системы можно было восстановить данные из последней рабочей копии.

4. Как создаётся файловая система?

- **Использовать команду mkfs:** В большинстве дистрибутивов Linux для создания файловой системы используется команда mkfs, например, для создания ext4-файловой системы: `mkfs.ext4 /dev/sdX`.
- **Настройка параметров файловой системы:** Вы можете указать различные параметры, такие как размер блока, метку тома и другие опции в зависимости от вашего выбора файловой системы.
- **Монтирование файловой системы (опционально):** После создания файловой системы вы можете выполнить операцию монтирования, чтобы сделать её доступной для использования операционной системой.

5. Дайте характеристику командам для просмотра текстовых файлов.

Команды для просмотра текстовых файлов предоставляют пользователю возможность просматривать содержимое текстовых файлов непосредственно в терминале или командной строке. Вот несколько основных команд для просмотра текстовых файлов:

- **cat:** Команда cat (concatenate) используется для отображения содержимого одного или нескольких файлов в терминале. Она также может использоваться для объединения нескольких файлов и вывода содержимого на экран.

- **less:** Команда `less` позволяет просматривать содержимое файла постранично. Это удобно для просмотра больших файлов, так как позволяет легко прокручивать вперед и назад, а также искать в тексте.
- **more:** Подобно команде `less`, команда `more` также используется для просмотра содержимого файла постранично. Однако `more` менее гибкая по сравнению с `less`.
- **head:** Команда `head` выводит начальные строки текстового файла. По умолчанию выводится 10 первых строк файла, но это количество можно изменить с помощью параметра.
- **tail:** Команда `tail` выводит конечные строки текстового файла. По умолчанию выводится 10 последних строк файла, но это количество также можно изменить с помощью параметра.

6. Приведите основные возможности команды `cp` в Linux.

Команда `cp` в Linux предназначена для копирования файлов и каталогов. Она имеет ряд опций, позволяющих выполнять различные операции с копированием. Вот основные возможности команды `cp`:

- **Копирование файлов:** С помощью `cp` можно скопировать один или несколько файлов из одного места в другое. Например: `cp file1.txt /путь/к/директории`.
- **Копирование каталогов:** Команда `cp` также позволяет копировать целые каталоги со всем их содержимым. Для этого используется опция `-r` (рекурсивное копирование).
- **Подтверждение при копировании:** При копировании файлов в уже существующий каталог с тем же именем, команда `cp` запрашивает подтверждение, если вы не используете опцию перезаписи.

- **Переименование файлов:** Можно использовать команду `cp` для переименования файлов, скопировав файл с новым именем. Например: `cp file1.txt newfile.txt`.
- **Сохранение атрибутов файлов:** Опции `cp` позволяют сохранить атрибуты файлов, такие как права доступа, временные метки и владельца файла.
- **Рекурсивное копирование:** С помощью опции `-r` или `-R` можно скопировать каталог и всё его содержимое, включая подкаталоги.
- **Создание резервной копии:** Команда `cp` может быть использована для создания резервных копий файлов и каталогов.
- **Копирование множества файлов в один каталог:** Можно одновременно скопировать несколько файлов в один каталог, указав последним аргументом имя каталога.

7. Приведите основные возможности команды `mv` в Linux.

Команда `mv` в Linux используется для перемещения файлов и каталогов. Она также может быть использована для переименования файлов и каталогов. Вот основные возможности команды `mv`:

- **Перемещение файлов:** Команда `mv` позволяет переместить один или несколько файлов из одной директории в другую. Например: `mv file1.txt /путь/к/другой_директории`.
- **Перемещение каталогов:** Аналогично файлам, команда `mv` может перемещать целые каталоги со всем их содержимым.
- **Переименование файлов:** Путем использования `mv` можно также переименовывать файлы, перемещая файл с новым именем. Например: `mv oldname.txt newname.txt`.

- **Перезапись файлов:** Если файл с указанным именем уже существует в целевой директории, команда `mv` перезапишет его без предупреждения.
- **Сохранение атрибутов файлов:** При перемещении файлов команда `mv` сохраняет атрибуты файлов, такие как права доступа, временные метки и владельца файла.
- **Перемещение и переименование:** Команда `mv` может одновременно перемещать файлы и переименовывать их, предоставляя гибкость в управлении файлами и директориями.
- **Безопасное перемещение множества файлов в один каталог:** Можно одновременно переместить несколько файлов в один каталог, указав последним аргументом имя каталога.
- **Перемещение с подтверждением:** При перемещении файлов в уже существующий каталог с тем же именем, команда `mv` запрашивает подтверждение, если вы не используете опцию перезаписи.

8. Что такое права доступа? Как они могут быть изменены?

Права доступа определяют, какие действия могут выполнять пользователи и программы в отношении файлов и каталогов. В системах Unix-подобных операционных системах (таких как Linux), каждый файл и каталог имеет набор прав доступа, который определяет, кто может читать, записывать или выполнять файл, а также кто может изменять его.

Вот основные типы прав доступа:

- **Чтение (Read):** Пользователь или программа может просматривать содержимое файла или каталога.
- **Запись (Write):** Пользователь или программа может изменять файл или каталог, добавлять или удалять содержимое.

- **Выполнение (Execute):** Для каталогов это позволяет входить в каталог и выполнять команды внутри него. Для файлов это позволяет запускать исполняемые файлы.

Права доступа могут быть изменены с помощью команды `chmod` в терминале Linux. Вот основные способы изменения прав доступа:

- **Использование символьной нотации:** Например, `chmod u+r file.txt` добавляет право чтения для владельца файла.
- **Использование числовой нотации:** Назначение прав осуществляется через числовые значения, такие как 755, где каждая цифра представляет набор прав для владельца, группы и других пользователей соответственно.
- **Изменение прав доступа по битам:** Используя флаги `u`, `g`, `o` и комбинации `+` и `-` можно добавлять или удалять права доступа.

6 Выводы

В данной лабораторной работе мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов, а также приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

Список литературы

1. Руководство к лабораторной работе №4.