

# **Отчёт по лабораторной работе №5**

**дисциплина: Операционные системы**

Сычев Егор Олегович

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	18
4	Контрольные вопросы	19

## Список иллюстраций

2.1	touch, cp . . . . .	6
2.2	mkdir, cp, ls . . . . .	7
2.3	mv, ls, mkdir . . . . .	8
2.4	touch, ls, mkdir, chmod . . . . .	9
2.5	cp, mv, ls, mkdir . . . . .	10
2.6	touch, cp, mv, ls, mkdir . . . . .	10
2.7	chmod . . . . .	11
2.8	chmod . . . . .	11
2.9	chmod, ls . . . . .	11
2.10	cat . . . . .	12
2.11	cp, mv . . . . .	12
2.12	mv, chmod, cat, cp . . . . .	13
2.13	man mount . . . . .	14
2.14	man fsck . . . . .	15
2.15	man mkfs . . . . .	16
2.16	man kill . . . . .	17

## **Список таблиц**

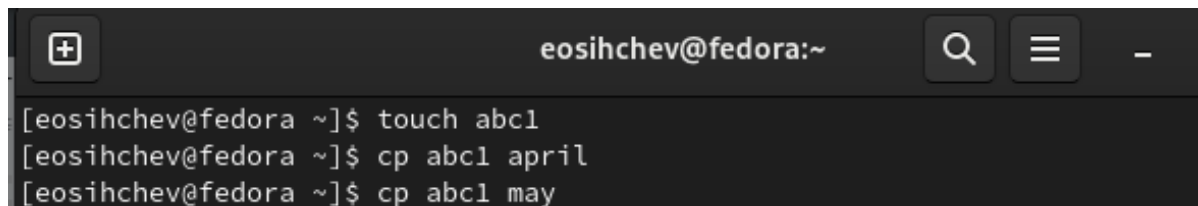
# 1 Цель работы

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

## 2 Выполнение лабораторной работы

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

- Создаем файл abc1
- Копируем файл abc1 в файл april и may
- Создаем каталог monthly
- Копируем файлы april и may в каталог monthly
- Копируем файл may в файл june
- Создаем каталог monthly.00
- Копируем каталог monthly.00 в каталог monthly
- Копируем каталог monthly.00 в каталог tmp

A terminal window with a dark background. The title bar shows a plus icon, the username 'eosihchev@fedora:~', a search icon, a menu icon, and a close icon. The terminal content shows three lines of commands and their prompts: '[eosihchev@fedora ~]\$ touch abc1', '[eosihchev@fedora ~]\$ cp abc1 april', and '[eosihchev@fedora ~]\$ cp abc1 may'.

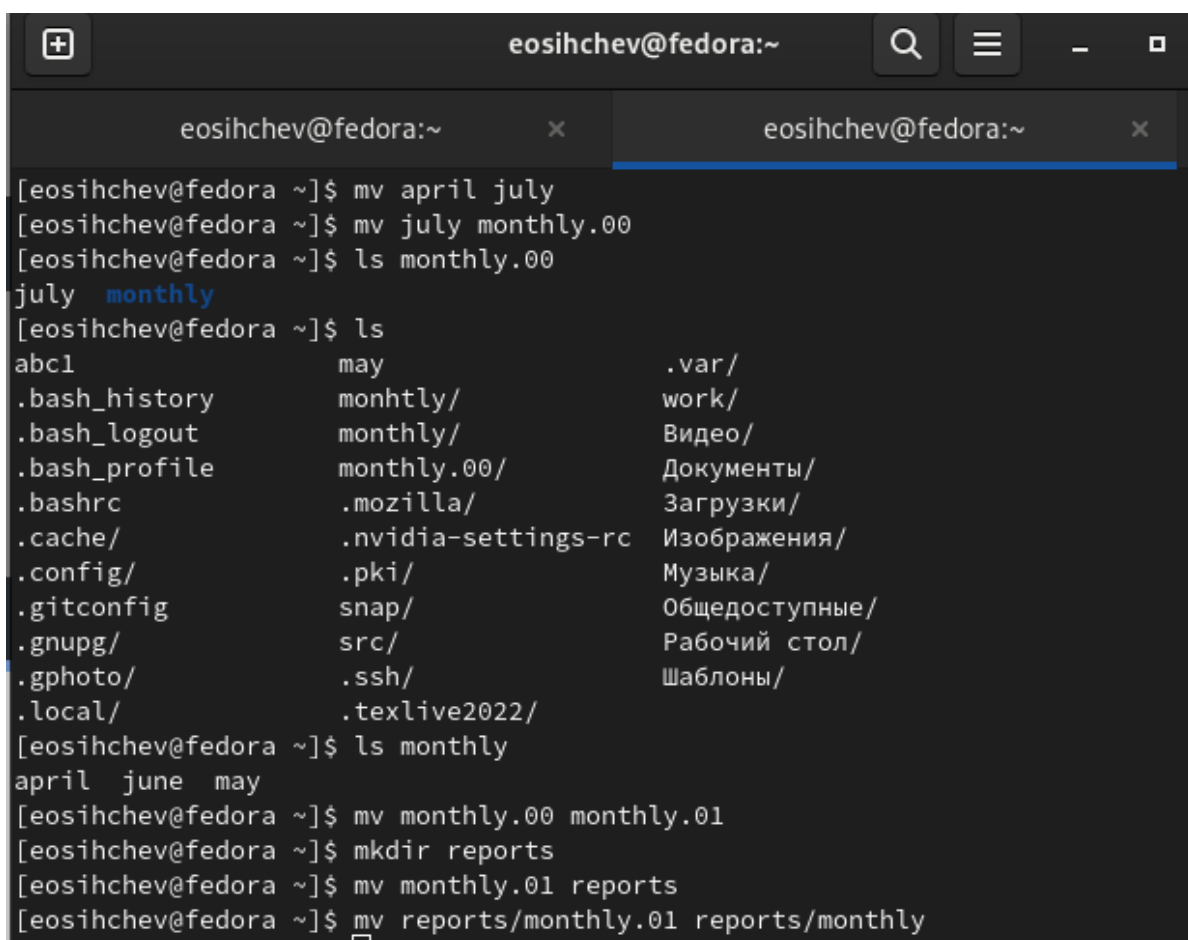
```
[eosihchev@fedora ~]$ touch abc1
[eosihchev@fedora ~]$ cp abc1 april
[eosihchev@fedora ~]$ cp abc1 may
```

Рис. 2.1: touch, cp

```
[eosihchev@fedora ~]$ mkdir monthly
[eosihchev@fedora ~]$ cp april may monthly
[eosihchev@fedora ~]$ cp monthly/may monthly/june
[eosihchev@fedora ~]$ ls monthly
april  june  may
[eosihchev@fedora ~]$ mkdir monthly.00
[eosihchev@fedora ~]$ cp -r monthly monthly.00
[eosihchev@fedora ~]$ cp -r monthly.00 /tmp
[eosihchev@fedora ~]$
```

Рис. 2.2: mkdir, cp, ls

- Изменяем название файла april на july
- Перемещаем файл jult в каталог monthly.00
- Переименуем каталог monthly.00 в monthly.01
- Создаем каталог reports
- Перемещаем каталог monthly.01 в каталог reports
- Переименуем каталог reports/monthly.01 в monthly

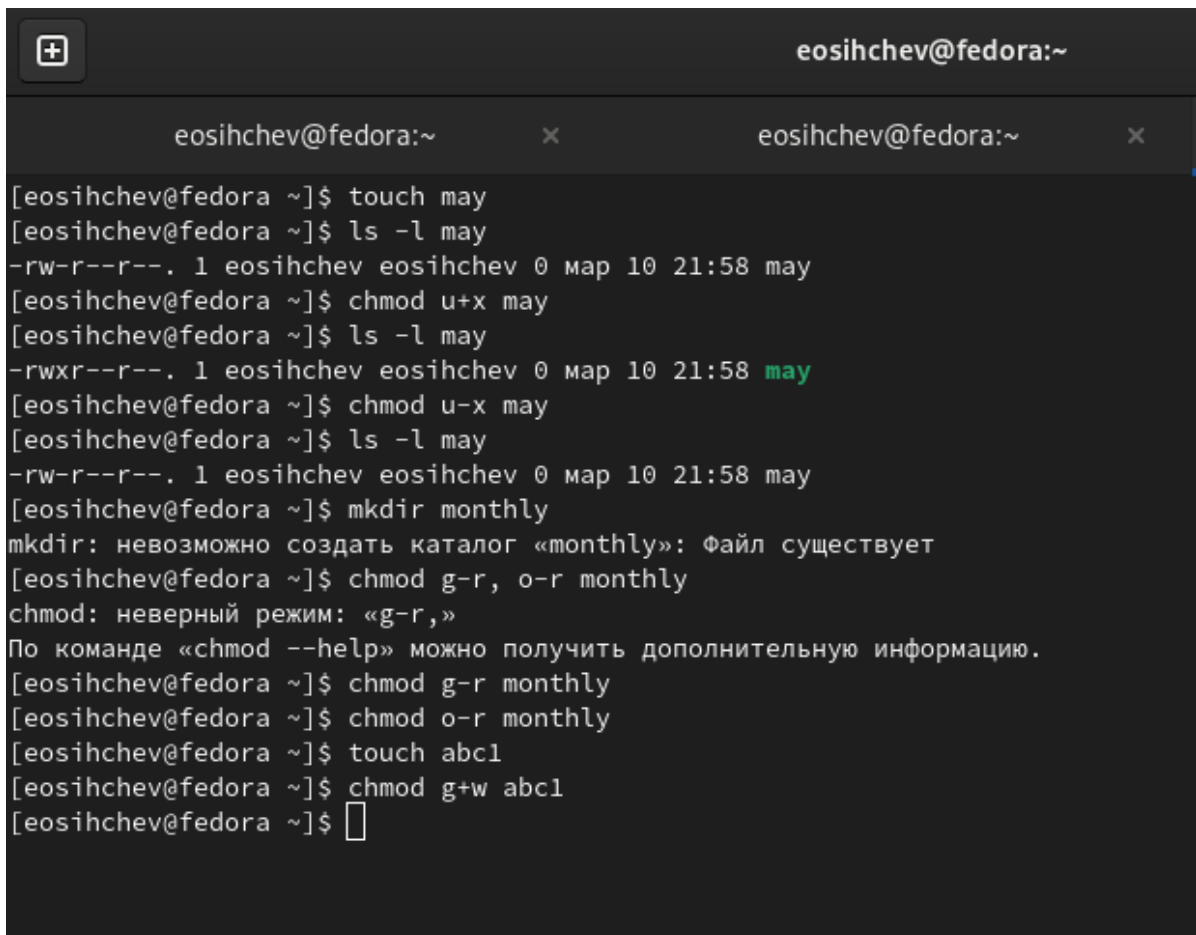


```
eosihchev@fedora:~  
[eosihchev@fedora ~]$ mv april july  
[eosihchev@fedora ~]$ mv july monthly.00  
[eosihchev@fedora ~]$ ls monthly.00  
july  monthly  
[eosihchev@fedora ~]$ ls  
abcl      may          .var/  
.bash_history  monthtly/    work/  
.bash_logout  monthly/     Видео/  
.bash_profile  monthly.00/  Документы/  
.bashrc        .mozilla/    Загрузки/  
.cache/        .nvidia-settings-rc  Изображения/  
.config/       .pki/        Музыка/  
.gitconfig     snap/        Общедоступные/  
.gnupg/        src/         Рабочий стол/  
.gphoto/       .ssh/        Шаблоны/  
.local/        .texlive2022/  
[eosihchev@fedora ~]$ ls monthly  
april  june  may  
[eosihchev@fedora ~]$ mv monthly.00 monthly.01  
[eosihchev@fedora ~]$ mkdir reports  
[eosihchev@fedora ~]$ mv monthly.01 reports  
[eosihchev@fedora ~]$ mv reports/monthly.01 reports/monthly
```

Рис. 2.3: mv, ls, mkdir

- Создаем файл may с правом выполнения для владельца
- Лишаем владельца файла may права на выполнение
- Создаем каталог monthly с запретом на чтение
- Создаем файл abcl с правом записи





```
eosihchev@fedora:~  
eosihchev@fedora:~ x eosihchev@fedora:~ x  
[eosihchev@fedora ~]$ touch may  
[eosihchev@fedora ~]$ ls -l may  
-rw-r--r--. 1 eosihchev eosihchev 0 map 10 21:58 may  
[eosihchev@fedora ~]$ chmod u+x may  
[eosihchev@fedora ~]$ ls -l may  
-rwxr--r--. 1 eosihchev eosihchev 0 map 10 21:58 may  
[eosihchev@fedora ~]$ chmod u-x may  
[eosihchev@fedora ~]$ ls -l may  
-rw-r--r--. 1 eosihchev eosihchev 0 map 10 21:58 may  
[eosihchev@fedora ~]$ mkdir monthly  
mkdir: невозможно создать каталог «monthly»: Файл существует  
[eosihchev@fedora ~]$ chmod g-r, o-r monthly  
chmod: неверный режим: «g-r,»  
По команде «chmod --help» можно получить дополнительную информацию.  
[eosihchev@fedora ~]$ chmod g-r monthly  
[eosihchev@fedora ~]$ chmod o-r monthly  
[eosihchev@fedora ~]$ touch abc1  
[eosihchev@fedora ~]$ chmod g+w abc1  
[eosihchev@fedora ~]$
```

Рис. 2.4: touch, ls, mkdir, chmod

## 2. 2.1 - 2.8

- Копируем файл io.h в домашний каталог и называем его equipment
- Создаем директорию ski.plases
- Перемещаем файл equipment в каталог ski.plases
- Переименоуем файл ski.plasese в equiplist
- Создаем файл abc1 и копируем его в каталог ski.plases
- Поменяем название файла abc1 на equiplist2
- Создаем каталог equipment в каталоге ski.plases
- Перемещаем файлы equiplist и equiplist2 в каталоге equipment
- Создаем и перемещаем каталог newdir в каталог ski.plases и называем его plans

```
eosihchev@fedora:~  
eosihchev@fedora:~ x eosihchev@fedora:~ x eosihchev@fedora:~ x eosihchev@fedora:~ x  
[eosihchev@fedora ~]$ cp /usr/include/sys/io.h ~  
[eosihchev@fedora ~]$ mv io.h equipment  
[eosihchev@fedora ~]$ ls  
abc1    may    monthly  snap    work    Документы  Изображения  Общедоступные  Шаблоны  
equipment  monthly  reports  src      Видео    Загрузки    Музыка    'Рабочий стол'  
[eosihchev@fedora ~]$ mkdir ski.plases  
[eosihchev@fedora ~]$ mv equipment ski.plases/  
bash: mv: команда не найдена...  
[eosihchev@fedora ~]$ mv equipment ski.plases/  
[eosihchev@fedora ~]$ mv ~/ski.plases/equipment ~/ski.plases/equiplist  
[eosihchev@fedora ~]$ ls ski.plases/  
equiplist  
[eosihchev@fedora ~]$
```

Рис. 2.5: cp, mv, ls, mkdir

```
eosihchev@fedora:~  
eosihchev@fed... x eosihchev@fed... x eosihchev@fed... x eosihchev@fed... x eosihchev@fed... x  
[eosihchev@fedora ~]$ touch abc1  
[eosihchev@fedora ~]$ cp abc1 ski.plases/  
[eosihchev@fedora ~]$ mv ski.plases/abc1 ski.plases/equiplist2  
[eosihchev@fedora ~]$ ls ski.plases/  
equiplist  equiplist2  
[eosihchev@fedora ~]$ mkdir newdir  
[eosihchev@fedora ~]$ mv newdir/ plans  
[eosihchev@fedora ~]$ mv plans ski.plases/  
[eosihchev@fedora ~]$ ls  
abc1    monthly  reports  snap    work    Документы  Изображения  Общедоступные  Шаблоны  
may    monthly  ski.plases  src      Видео    Загрузки    Музыка    'Рабочий стол'  
[eosihchev@fedora ~]$ mkdir australia  
[eosihchev@fedora ~]$ mkdir play  
[eosihchev@fedora ~]$ touch my_os  
[eosihchev@fedora ~]$ touch feathers  
[eosihchev@fedora ~]$
```

Рис. 2.6: touch, cp, mv, ls, mkdir

### 3. 3.1 - 3.4

- 3.1: chmod 744 australia
- 3.2: chmod 711 play
- 3.3: chmod 544 my\_os
- 3.4: chmod 664 feathers

```
eosihchev@fedora:~  
eosihchev... x eosihchev... x eosihchev... x eosihchev... x eosihchev... x eosihchev... x  
[eosihchev@fedora ~]$ chmod 744 australia/  
[eosihchev@fedora ~]$  
[eosihchev@fedora ~]$ chmod 544 my_os
```

Рис. 2.7: chmod

```
[eosihchev@fedora ~]$ chmod 711 play
```

Рис. 2.8: chmod

```
[eosihchev@fedora ~]$ chmod 664 feathers  
[eosihchev@fedora ~]$ ls -l  
итого 0  
-rw-rw-r--. 1 eosihchev eosihchev  0 мар 10 22:10  abcl  
drwxr--r--. 1 eosihchev eosihchev  0 мар 10 22:20  australia  
-rw-rw-r--. 1 eosihchev eosihchev  0 мар 10 22:20  feathers  
-rw-r--r--. 1 eosihchev eosihchev  0 мар 10 21:58  may  
drwxr-xr-x. 1 eosihchev eosihchev  0 мар 10 21:33  monthtly  
drwx--x--x. 1 eosihchev eosihchev 24 мар 10 21:34  monthly  
-r-xr--r--. 1 eosihchev eosihchev  0 мар 10 22:20  my_os  
drwx--x--x. 1 eosihchev eosihchev  0 мар 10 22:20  play  
drwxr-xr-x. 1 eosihchev eosihchev 14 мар 10 21:53  reports  
drwxr-xr-x. 1 eosihchev eosihchev 48 мар 10 22:18  ski.places  
drwx-----. 1 eosihchev eosihchev 18 фев 25 17:04  snap  
drwxrwxr-x. 1 eosihchev eosihchev 30 окт 26 05:35  src  
drwxr-xr-x. 1 eosihchev eosihchev 70 фев 25 17:45  work  
drwxr-xr-x. 1 eosihchev eosihchev 46 мар 10 21:32  Видео  
drwxr-xr-x. 1 eosihchev eosihchev  0 окт 26 04:16  Документы  
drwxr-xr-x. 1 eosihchev eosihchev 982 фев 25 20:30  Загрузки  
drwxr-xr-x. 1 eosihchev eosihchev 66 фев 25 19:21  Изображения  
drwxr-xr-x. 1 eosihchev eosihchev  0 окт 26 04:16  Музыка  
drwxr-xr-x. 1 eosihchev eosihchev  0 окт 26 04:16  Общедоступные  
drwxr-xr-x. 1 eosihchev eosihchev  0 окт 26 04:16  'Рабочий стол'  
drwxr-xr-x. 1 eosihchev eosihchev  0 окт 26 04:16  Шаблоны  
[eosihchev@fedora ~]$
```

Рис. 2.9: chmod, ls

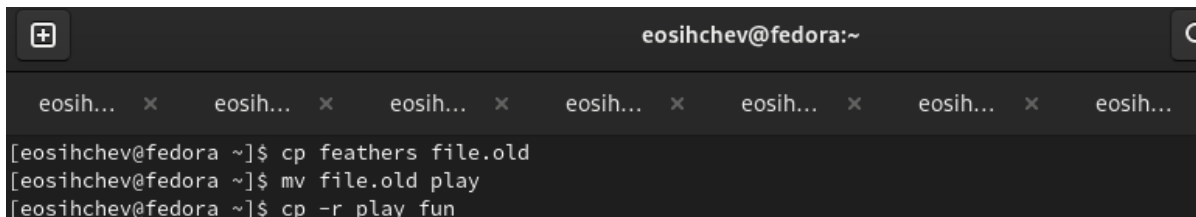
#### 4. 4.1 - 4.12

- Смотрим содержимое файла passwd

```
[eosihchev@fedora ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/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:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
games:x:12:100:games:/usr/games:/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/:/sbin/nologin
dbus:x:81:81:System message bus:/:/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
tss:x:59:59:Account used for TPM access:/dev/null:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:usr/sbin/nologin
systemd-oom:x:999:999:systemd Userspace OOM Killer:/:usr/sbin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:usr/sbin/nologin
qemu:x:107:107:qemu user:/:/sbin/nologin
polkitd:x:998:997:User for polkitd:/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
unbound:x:997:995:Unbound DNS resolver:/etc/unbound:/sbin/nologin
nm-openconnect:x:996:994:NetworkManager user for OpenConnect:/:/sbin/nologin
```

Рис. 2.10: cat

- Копируем файл feathers в файл file.old
- Перемещаем файл file.old в каталог play
- Копируем каталог play в каталог fun



```
eosihchev@fedora:~
[eosihchev@fedora ~]$ cp feathers file.old
[eosihchev@fedora ~]$ mv file.old play
[eosihchev@fedora ~]$ cp -r play fun
```

Рис. 2.11: cp, mv

- Перемещаем каталог fun в каталог play
- Лишаем владельца файла feathers права чтения
- Получим отказ в доступе
- Получим отказ в доступе
- Даем владельцу файла feathers право на чтение

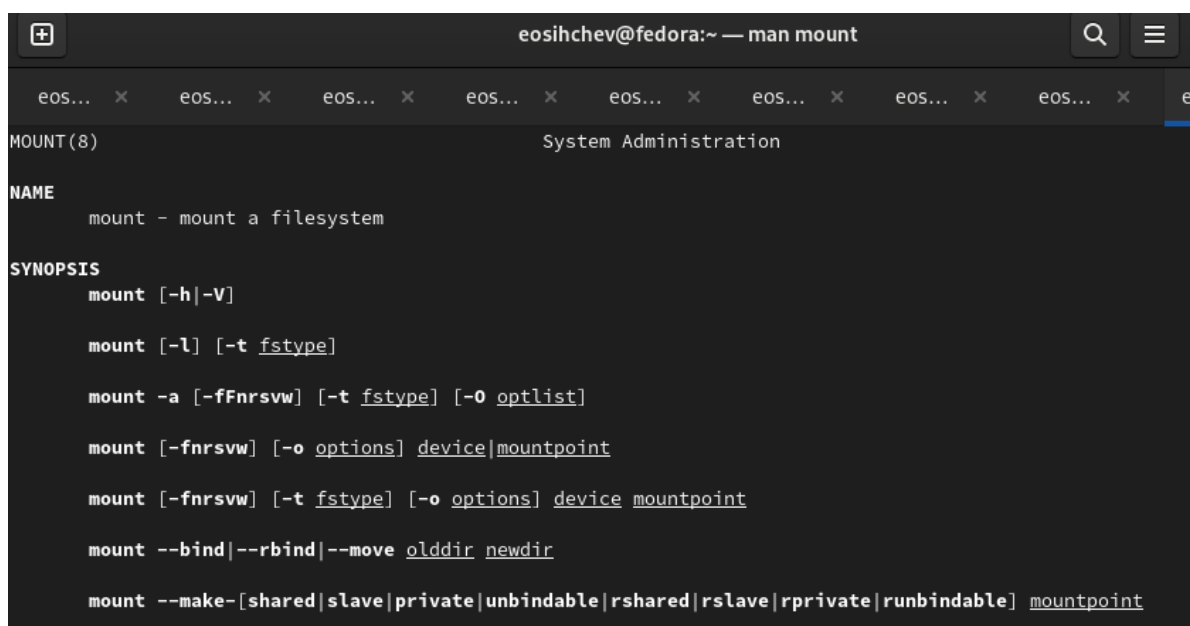
- Лишаем владельца каталога play право на выполнение
- Получим отказ в доступе
- Даем владельцу каталога play право на выполнение

```
[eosihchev@fedora ~]$ mv fun play
[eosihchev@fedora ~]$ mv play/fun play/games
[eosihchev@fedora ~]$ chmod u-r feathers
[eosihchev@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[eosihchev@fedora ~]$ cp feathers monthly/
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[eosihchev@fedora ~]$ chmod u+r feathers
[eosihchev@fedora ~]$ chmod u-x play/
[eosihchev@fedora ~]$
```

Рис. 2.12: mv, chmod, cat, cp

## 5. Команды mount, fsck, mkfs, kill

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.



```
eosihchev@fedora:~ — man mount
MOUNT(8)                                System Administration

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
```

Рис. 2.13: man mount

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»

```
eosihchev@fedora:~ — man fsck
FCK(8) System Administration FCK(8)

NAME
    fsck - check and repair a Linux filesystem

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

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

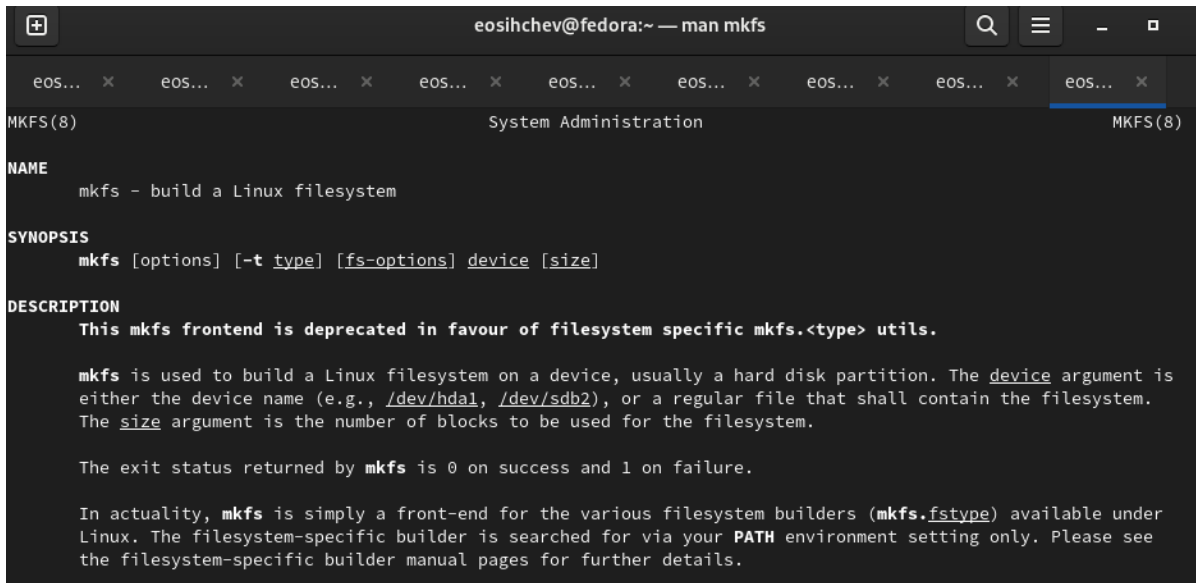
    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default
    to checking filesystems in /etc/fstab serially. This is equivalent to the -As options.

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

    0
    No errors
```

Рис. 2.14: man fsck

mkfs создаёт новую файловую систему Linux. Имеет следующий синтаксис: `mkfs -V -t fstype fs-options filesys blocks` mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesys для файловой системы может выступать или название устройства (например, `/dev/hda1`, `/dev/sdb2`) или точка монтирования (например, `/`, `/usr`, `/home`).



```
eosihchev@fedora:~ — 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.
```

Рис. 2.15: man mkfs

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



```
eosihchev@fedora:~ — 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.
```

Рис. 2.16: man kill

## 3 Вывод

Я ознакомился с файловой системой Linux, её структурой, именами и содержанием каталогов и приобрел 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.

Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th» (Рисунок 15). Из рисунка видно, что на моем компьютере есть следующие файловые системы: devtmpfs, tmpfs, ext4, iso9660. 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 «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. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.

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

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

Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой mount

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

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. Как создаётся файловая система?

Файловую систему можно создать, используя команду `mkfs`. `mkfs`: создаёт новую файловую систему Linux. Имеет следующий синтаксис: `mkfs -V -t fstype fs-options filesys blocks` `mkfs` используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента `filesys` для файловой системы может выступать или название устройства (например, `/dev/hda1`, `/dev/sdb2`) или точка монтирования (например, `/`, `/usr`, `/home`)

6. Дайте характеристику командам для просмотра текстовых файлов.

Для просмотра текстовых файлов существуют следующие команды: 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 Команда `nl` действует аналогично команде `cat`, но выводит еще и номера строк в столбце слева. 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 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: -с (-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 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: -с – 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. Приведите основные возможности команды cp в Linux.

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. Приведите основные возможности команды mv в Linux.

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 -striptrailing-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. Что такое права доступа? Как они могут быть изменены?

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: = установить право лишить права дать право r чтение w запись x выполнение u (user) владелец файла g (group) группа, к которой принадлежит владелец файла o (others) все остальные