

Отчет по лабораторной работе №7

Операционные системы

Зоригоо Номун

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Выводы	21
6	Ответы на контрольные вопросы	22

Список иллюстраций

4.1	Создание файла	9
4.2	Создание директории	10
4.3	Копирование файла	10
4.4	Создание директории	11
4.5	Переименовывание файла	12
4.6	Создание директории	12
4.7	Работа с правами доступа	13
4.8	Работа с правами доступа	13
4.9	Работа с правами доступа	14
4.10	Проверка файловой системы	14
4.11	Копирование файла	15
4.12	Создание файла	15
4.13	Создание директории	16
4.14	Работа с правами доступа	16
4.15	Работа с правами доступа	17
4.16	Работа с правами доступа	17
4.17	Чтение файла	18
4.18	Копирование файла	18
4.19	Работа с правами доступа	19
4.20	Работа с правами доступа	19
4.21	Чтение документации	20

Список таблиц

1 Цель работы

Целью данной лабораторной работы является 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. Выполнить все примеры из лабораторной работы
2. Выполнить команды по копированию, созданию и перемещению файлов и каталогов
3. Определить опции команды `chmod`
4. Изменить права доступа к файлам
5. Прочитать документацию о командах `mount`, `fsck`, `mkfs`, `kill`

3 Теоретическое введение

Для создания текстового файла можно использовать команду `touch`. Для просмотра файлов небольшого размера можно использовать команду `cat`. Для просмотра файлов постранично удобнее использовать команду `less`. Команда `cp` используется для копирования файлов и каталогов. Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов.

Каждый файл или каталог имеет права доступа. В сведениях о файле или каталоге указываются:

- тип файла (символ (-) обозначает файл, а символ (d) — каталог);
- права для владельца файла (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует);
- права для членов группы (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует);
- права для всех остальных (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует).

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.

Файловая система в Linux состоит из файлов и каталогов. Каждому физическому носителю соответствует своя файловая система. Существует несколько типов файловых систем. Перечислим наиболее часто встречающиеся типы:

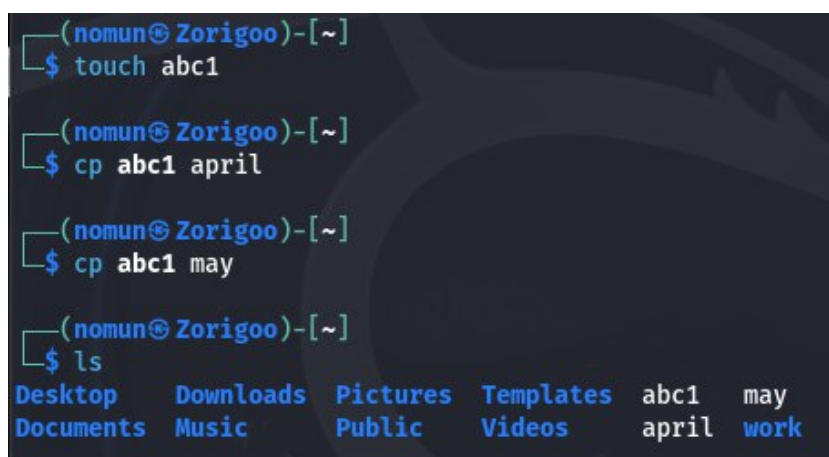
- ext2fs (second extended filesystem);
- ext3fs (third extended file system);

- ext4 (fourth extended file system);
- ReiserFS;
- xfs;
- fat (file allocation table);
- ntfs (new technology file system).

Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой `mount` без параметров.

4 Выполнение лабораторной работы

Создаю файл, дважды копирую его с новыми именами и проверяю, что все команды были выполнены корректно (рис. fig. 4.1).

A terminal window with a dark background and a faint dragon watermark. It shows four lines of commands and their outputs. The first line creates a file 'abc1'. The next two lines copy 'abc1' to 'april' and 'may'. The final line lists the contents of the current directory, showing standard Linux folders and the three files created.

```
(nomun@Zorigoo)-[~]  
$ touch abc1  
  
(nomun@Zorigoo)-[~]  
$ cp abc1 april  
  
(nomun@Zorigoo)-[~]  
$ cp abc1 may  
  
(nomun@Zorigoo)-[~]  
$ ls  
Desktop    Downloads  Pictures   Templates  abc1    may  
Documents  Music      Public     Videos    april   work
```

Рис. 4.1: Создание файла

Создаю директорию, копирую в нее два файла, созданных на прошлом этапе, проверяю, что все скопировалось (рис. fig. 4.2).

```
(nomun@Zorigoo)-[~]  
$ mkdir monthly  
  
(nomun@Zorigoo)-[~]  
$ cp april may monthly/  
  
(nomun@Zorigoo)-[~]  
$ ls monthly/  
april  may
```

Рис. 4.2: Создание директории

Копирую файл, находящийся не в текущей директории в файл с новым именем тоже не текущей директории (рис. fig. 4.3).

```
(nomun@Zorigoo)-[~]  
$ cp monthly/may monthly/june  
  
(nomun@Zorigoo)-[~]  
$ ls monthly/  
april  june  may
```

Рис. 4.3: Копирование файла

Создаю новую директорию. Копирую предыдущую созданную директорию вместе со всем содержимым в каталог /tmp. Затем копирую предыдущую созданную директорию в новую созданную (рис. fig. 4.4).

```
(nomun@Zorigoo)-[~]  
$ mkdir monthly.00  
  
(nomun@Zorigoo)-[~]  
$ cp -r monthly /tmp  
  
(nomun@Zorigoo)-[~]  
$ ls monthly  
april  june  may  
  
(nomun@Zorigoo)-[~]  
$ cp -r monthly monthly.00  
"3.png" selected (17.0 kB)  
  
(nomun@Zorigoo)-[~]  
$ ls monthly.00  
monthly
```

Рис. 4.4: Создание директории

Переименовываю файл, затем перемещаю его в каталог (рис. fig. 4.5).

```
(nomun@Zorigoo)-[~]  
$ mv april july  
  
(nomun@Zorigoo)-[~]  
$ mv july monthly.00  
"4.png" selected (31.9 kB)  
  
(nomun@Zorigoo)-[~]  
$ ls monthly.00/  
july  monthly
```

Рис. 4.5: Переименовывание файла

Создаю новую директорию, переименовываю monthly.00 в monthly.01, перемещаю директорию в директорию reports, переименовываю эту директорию, убираю из названия 01 (рис. fig. 4.6).

```
(nomun@Zorigoo)-[~]  
$ mkdir reports  
  
(nomun@Zorigoo)-[~]  
$ mv monthly.00 monthly.01  
  
(nomun@Zorigoo)-[~]  
$ mv monthly.01/ reports/  
  
(nomun@Zorigoo)-[~]  
$ mv reports/monthly.01 reports/monthly
```

Рис. 4.6: Создание директории

Создаю пустой файл, проверяю права доступа у него, изменяю права доступа, добавляя пользователю (создателю) можно выполнять файл (рис. fig. 4.7).

```
(nomun@Zorigoo)-[~]
$ touch may

(nomun@Zorigoo)-[~]
$ ls -l may
-rw-r--r-- 1 nomun nomun 0 Mar 23 16:14 may

(nomun@Zorigoo)-[~]
$ chmod u+x may

(nomun@Zorigoo)-[~]
$ ls -l may
-rwxr--r-- 1 nomun nomun 0 Mar 23 16:14 may

(nomun@Zorigoo)-[~]
$ chmod u-x may

(nomun@Zorigoo)-[~]
$ ls -l may
-rw-r--r-- 1 nomun nomun 0 Mar 23 16:14 may
```

Рис. 4.7: Работа с правами доступа

Меняю права доступа у директории: группы и остальные пользователи не смогут ее прочесть (рис. fig. 4.8).

```
(nomun@Zorigoo)-[~]
$ chmod g-r monthly

(nomun@Zorigoo)-[~]
$ chmod o-r monthly
```

Рис. 4.8: Работа с правами доступа

Изменяю права доступа у директории, запрещаю группам и остальным пользователям читать. Создаю новый пустой файл, даю ему права доступа: группы могут в этом чате писать содержимое (рис. fig. 4.9).

```
(nomun@Zorigoo)-[~]
$ touch abc1

(nomun@Zorigoo)-[~]
$ chmod g+w abc1

(nomun@Zorigoo)-[~]
$ ls -l abc1
-rw-rw-r-- 1 nomun nomun 0 Mar 23 16:22 abc1
```

Рис. 4.9: Работа с правами доступа

Проверяю файловую систему (рис. fig. 4.10).

```
(nomun@Zorigoo)-[~]
$ fsck /dev/sda1
fsck from util-linux 2.39.3
e2fsck 1.47.0 (5-Feb-2023)
/dev/sda1 is mounted.

WARNING!!! The filesystem is mounted.  If you continue you ***WILL***
cause ***SEVERE*** filesystem damage.

Do you really want to continue<n>? yes
fsck.ext4: Permission denied while trying to open /dev/sda1
You must have r/w access to the filesystem or be root

(nomun@Zorigoo)-[~]
$ sudo fsck /dev/sda1
[sudo] password for nomun:
fsck from util-linux 2.39.3
e2fsck 1.47.0 (5-Feb-2023)
/dev/sda1 is mounted.
```

Рис. 4.10: Проверка файловой системы

Копирую файл в домашний каталог с новым именем, создаю новую пустую директорию, перемещаю файл в эту директорию, переименовываю файл (рис. fig. 4.11).

```

(nomun@Zorigoo)-[~]
$ mv equipment ski.places/equipment

(nomun@Zorigoo)-[~]
$ ls
Desktop  Downloads  Pictures  Templates  abc1  monthly  ski.places
Documents Music      Public    Videos    may   reports  work

(nomun@Zorigoo)-[~]
$ ls ski.places
equipment

(nomun@Zorigoo)-[~]
$ mv ski.places/equipment ski.places/equiplist

(nomun@Zorigoo)-[~]
$ ls ski.places
equiplist

```

Рис. 4.11: Копирование файла

Создаю новый файл, копирую его в новую директорию, но уже сразу с новым именем. Создаю внутри этого каталога подкаталог, перемещаю файлы в подкаталог (рис. fig. 4.12).

```

(nomun@Zorigoo)-[~]
$ touch abc1

(nomun@Zorigoo)-[~]
$ cp abc1 ski.places/equiplist2

(nomun@Zorigoo)-[~]
$ mkdir ski.places/equipment
mkdir: cannot create directory 'ski.places/equipment': File exists

(nomun@Zorigoo)-[~]
$ mv ski.places/equiplist ski.places/equiplist2 ski.places/equipment/
mv: cannot stat 'ski.places/equiplist': No such file or directory

(nomun@Zorigoo)-[~]
$ cd ski.places

(nomun@Zorigoo)-[~/ski.places]
$ ls
equipment

(nomun@Zorigoo)-[~/ski.places]
$ mkdir equiplist2

(nomun@Zorigoo)-[~/ski.places]
$ ls
equiplist2  equipment

```

Рис. 4.12: Создание файла

Создаю новую директорию, в этой же строчке перемещаю ее с новым именем в директорию, созданную в прошлый раз (рис. fig. 4.13).

```
(nomun@Zorigoo)-[~]
$ mkdir newdir ; mv newdir ski.places/plans

(nomun@Zorigoo)-[~]
$ ls ski.places
equiplist2  equipment  plans
```

Рис. 4.13: Создание директории

Проверяю, какие права нужно поменять и как, чтобы у новой директории были нужные по заданию права. (рис. fig. 4.14).

```
(nomun@Zorigoo)-[~]
$ mkdir australia

(nomun@Zorigoo)-[~]
$ chmod u+x australia

(nomun@Zorigoo)-[~]
$ ls -l
total 52
drwxr-xr-x 2 nomun nomun 4096 Mar 23 15:45 Desktop
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Documents
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Downloads
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Music
drwxr-xr-x 3 nomun nomun 4096 Mar 23 16:08 Pictures
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Public
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Templates
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Videos
-rw-r--r-- 1 nomun nomun 0 Mar 23 16:53 ab
-rw-rw-r-- 1 nomun nomun 0 Mar 23 16:53 abc1
drwxr-xr-x 2 nomun nomun 4096 Mar 23 16:56 australia
-rw-r--r-- 1 nomun nomun 0 Mar 23 16:14 may
drwx--x--x 2 nomun nomun 4096 Mar 23 16:10 monthly
drwxr-xr-x 3 nomun nomun 4096 Mar 23 16:13 reports
drwxr-xr-x 5 nomun nomun 4096 Mar 23 16:55 ski.places
drwxr-xr-x 4 nomun nomun 4096 Mar 23 15:58 work
```

Рис. 4.14: Работа с правами доступа

Проверяю, какие права нужно поменять и как, чтобы у новых файлов были нужные по заданию права. (рис. fig. 4.15).


```
(nomun@Zorigoo)-[~]
$ mkdir play

(nomun@Zorigoo)-[~]
$ chmod u+x play

(nomun@Zorigoo)-[~]
$ chnod g-r+x play/
Command 'chnod' not found, did you mean:
  command 'chmod' from deb coreutils
Try: sudo apt install <deb name>

(nomun@Zorigoo)-[~]
$ chmod g-r+x play/

(nomun@Zorigoo)-[~]
$ chmod o-r+x play/

(nomun@Zorigoo)-[~]
$ ls -l
total 56
drwxr-xr-x 2 nomun nomun 4096 Mar 23 15:45 Desktop
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Documents
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Downloads
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Music
drwxr-xr-x 3 nomun nomun 4096 Mar 23 16:08 Pictures
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Public
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Templates
drwxr-xr-x 2 nomun nomun 4096 Mar 23 14:42 Videos
-rw-r--r-- 1 nomun nomun  0 Mar 23 16:53 ab
-rw-rw-r-- 1 nomun nomun  0 Mar 23 16:53 abc1
drwxr-xr-x 2 nomun nomun 4096 Mar 23 16:56 australia
-rw-r--r-- 1 nomun nomun  0 Mar 23 16:14 may
drwx--x--x 2 nomun nomun 4096 Mar 23 16:10 monthly
drwx--x--x 2 nomun nomun 4096 Mar 23 16:57 play
```

Рис. 4.15: Работа с правами доступа

Создаю файл, добавляю в правах доступа право на исполнение и убираю право на запись для владельца, затем создаю следующий файл, ему в правах доступа добавляю право на запись для группы (рис. 4.16).

```
(nomun@Zorigoo)-[~]
$ touch my_os

(nomun@Zorigoo)-[~]
$ chmod u+x-w my_os

(nomun@Zorigoo)-[~]
$ ls -l my_os
-r-xr--r-- 1 nomun nomun 0 Mar 23 16:58 my_os

(nomun@Zorigoo)-[~]
$ touch feathers

(nomun@Zorigoo)-[~]
$ chmod g+w feathers

(nomun@Zorigoo)-[~]
$ ls -l feathers
-rw-rw-r-- 1 nomun nomun 0 Mar 23 16:59 feathers
```

Рис. 4.16: Работа с правами доступа

Читаю содержимое файл (рис. fig. 4.17).

```
(nomun@Zorigoo)-[~]  
$ cat /etc/passwd  
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
```

Рис. 4.17: Чтение файла

Копирую файл с новым именем, перемещаю его в ранее созданную директорию, рекурсивно ее копирую с новым именем, рекурсивно копирую в нее скопированную до этого папку (рис. fig. 4.18).

```
(nomun@Zorigoo)-[~]  
$ cp feathers file.old  
  
(nomun@Zorigoo)-[~]  
$ mv file.old play/file.old  
  
(nomun@Zorigoo)-[~]  
$ cp -r play/ fun/  
  
(nomun@Zorigoo)-[~]  
$ cp -r fun/ play/games  
  
(nomun@Zorigoo)-[~]  
$ ls play/  
file.old games
```

Рис. 4.18: Копирование файла

Убираю право на чтение у файла для создателя, поэтому не могу его прочесть, также не могу его скопировать, потому что отказано в доступе на чтение, возвращаю все права (рис. fig. 4.19).

```

(nomun@Zorigoo)-[~]
$ chmod u-r feathers

(nomun@Zorigoo)-[~]
$ ls -l feathers
--w-rw-r-- 1 nomun nomun 0 Mar 23 16:59 feathers

(nomun@Zorigoo)-[~]
$ cat feathers
cat: feathers: Permission denied

(nomun@Zorigoo)-[~]
$ cp feathers feathers1
cp: cannot open 'feathers' for reading: Permission denied

(nomun@Zorigoo)-[~]
$ chmod u+r feathers

```

Рис. 4.19: Работа с правами доступа

Убираю у директории право на исполнение для пользователя, пытаюсь в нее войти - отказано в доступе, возвращаю все права (рис. fig. 4.20).

```

(nomun@Zorigoo)-[~]
$ chmod u-x play/

(nomun@Zorigoo)-[~]
$ cd play/
cd: permission denied: play/

(nomun@Zorigoo)-[~]
$ chmod u+x play/

(nomun@Zorigoo)-[~]
$ cd play/

```

Рис. 4.20: Работа с правами доступа

Я прочитала описание каждой из четырех команд с помощью `man` (рис. fig. 4.21).
- `mount` — утилита командной строки в UNIX-подобных операционных системах. Применяется для монтирования файловых систем. - `fsck` (проверка файловой системы) - 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. - `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 - Команда `Kill` посылает указанный сигнал указанному процессу. Если не указано ни одного сигнала, посылается сигнал `SIGTERM`. Сигнал `SIGTERM` завершает лишь те процессы, которые не обрабатывают его приход. Для других процессов может быть необходимым послать сигнал `SIGKILL`, поскольку этот сигнал перехватить невозможно.



```
(nomun@Zorigoo)-[~/play]
$ man mount

(nomun@Zorigoo)-[~/play]
$ man fsck

(nomun@Zorigoo)-[~/play]
$ man mkfs

(nomun@Zorigoo)-[~/play]
$ man kill
```

Рис. 4.21: Чтение документации

5 Выводы

При выполнении данной лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрела практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы

6 Ответы на контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу. Ext2, Ext3, Ext4 или Extended Filesystem - это стандартная файловая система для Linux. Она была разработана еще для Minix. Она самая стабильная из всех существующих, кодовая база изменяется очень редко и эта файловая система содержит больше всего функций. Версия ext2 была разработана уже именно для Linux и получила много улучшений. В 2001 году вышла ext3, которая добавила еще больше стабильности благодаря использованию журналирования. В 2006 была выпущена версия ext4, которая используется во всех дистрибутивах Linux до сегодняшнего дня. В ней было внесено много улучшений, в том числе увеличен максимальный размер раздела до одного экзабайта.

Btrfs или B-Tree File System - это совершенно новая файловая система, которая сосредоточена на отказоустойчивости, легкости администрирования и восстановления данных. Файловая система объединяет в себе очень много новых интересных возможностей, таких как размещение на нескольких разделах, поддержка подтомов, изменение размера на лету, создание мгновенных снимков, а также высокая производительность. Но многими пользователями файловая система Btrfs считается нестабильной. Тем не менее, она уже используется как файловая система по умолчанию в OpenSUSE и SUSE Linux.

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

/ — root каталог. Содержит в себе всю иерархию системы;

/bin — здесь находятся двоичные исполняемые файлы. Основные общие команды, хранящиеся отдельно от других программ в системе (прим.: pwd, ls, cat, ps);

/boot — тут расположены файлы, используемые для загрузки системы (образ initrd, ядро vmlinuz);

/dev — в данной директории располагаются файлы устройств (драйверов). С помощью этих файлов можно взаимодействовать с устройствами. К примеру, если это жесткий диск, можно подключить его к файловой системе. В файл принтера же можно написать напрямую и отправить задание на печать;

/etc — в этой директории находятся файлы конфигураций программ. Эти файлы позволяют настраивать системы, сервисы, скрипты системных демонов;

/home — каталог, аналогичный каталогу Users в Windows. Содержит домашние каталоги учетных записей пользователей (кроме root). При создании нового пользователя здесь создается одноименный каталог с аналогичным именем и хранит личные файлы этого пользователя;

/lib — содержит системные библиотеки, с которыми работают программы и модули ядра;

/lost+found — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;

/media — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию /media/cdrom;

/mnt — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;

/opt — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);

/proc — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;

/root — директория, которая содержит файлы и личные настройки суперпользователя;

/run — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;

/sbin — аналогично /bin содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;

/srv — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);

/sys — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;

/tmp — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очищается при перезагрузке;

/usr — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а

не системой. Содержимое доступно только для чтения (кроме root). Каталог имеет вторичную иерархию и похож на корневой;

/var — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в /var/log, кэш в /var/cache, очереди заданий в /var/spool/ и так далее.

3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе? Монтирование тома.
4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы? Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

Один блок адресуется несколькими `inode` (принадлежит нескольким файлам). Блок помечен как свободный, но в то же время занят (на него ссылается `inode`). Блок помечен как занятый, но в то же время свободен (ни один `inode` на него не ссылается). Неправильное число ссылок в `inode` (недостаток или избыток ссылающихся записей в каталогах). Несовпадение между размером файла и суммарным размером адресуемых `inode` блоков. Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы). “Потерянные” файлы (правильные `inode`, на которые не ссылаются записи каталогов). Недопустимые или неразмещенные номера `inode` в записях каталогов.

5. Как создаётся файловая система? `mkfs` - позволяет создать файловую систему Linux.
6. Дайте характеристику командам для просмотра текстовых файлов. `Cat` - выводит содержимое файла на стандартное устройство вывода. Выполнение команды `head` выведет первые 10 строк текстового файла. Выполнение команды `tail` выведет последние 10 строк текстового файла. Команда `tac` - это тоже самое, что и `cat`, только отображает строки в обратном порядке. Для того, чтобы просмотреть огромный текстовый файл применяются команды для постраничного просмотра. Такие как `more` и `less`.
7. Приведите основные возможности команды `cp` в Linux. `Ср` – копирует или перемещает директорию, файлы.
8. Приведите основные возможности команды `mv` в Linux. `Mv` - переименовать или переместить файл или директорию
9. Что такое права доступа? Как они могут быть изменены? Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.