

Отчёт по лабораторной работе №5.
Анализ файловой системы Linux.
Команды для работы с файлами и
каталогами

Дисциплина: Операционные системы

Ганина Таисия Сергеевна, НКАбд-01-22

Содержание

1	Цель работы	6
2	Задание	7
3	Теоретическое введение	9
4	Выполнение лабораторной работы	11
5	Контрольные вопросы	26
6	Выводы	34
	Список литературы	35

Список иллюстраций

3.1	Права доступа	10
4.1	Скопировать файл ~/abc1 в файл april и в файл may	11
4.2	Скопировать файлы april и may в каталог monthly	11
4.3	Скопировать файл monthly/may в файл с именем june, скопировать каталог monthly в каталог monthly.00, скопировать каталог monthly.00 в каталог /tmp	12
4.4	Изменить название файла april на july в домашнем каталоге . . .	12
4.5	Переместить файл july в каталог monthly.00, переименовать каталог monthly.00 в monthly.01	12
4.6	Переместить каталог monthly.01 в каталог reports, переименовать каталог reports/monthly.01 в reports/monthly	12
4.7	Требуется создать файл ~/may с правом выполнения для владельца	13
4.8	Требуется лишить владельца файла ~/may права на выполнение .	13
4.9	Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей	13
4.10	Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей	13
4.11	Требуется создать файл ~/abc1 с правом записи для членов группы	13
4.12	Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой mount без параметров .	14
4.13	Другой способ определения смонтированных в операционной системе файловых систем — просмотр файла/etc/fstab	14
4.14	Для определения объёма свободного пространства на файловой системе можно воспользоваться командой df, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования	15
4.15	С помощью команды fsck можно проверить (а в ряде случаев восстановить) целостность файловой системы	15
4.16	Копирование файла	15
4.17	~/ski.places	16
4.18	Перемещая файл equipment в каталог ~/ski.places	16
4.19	~/ski.places/equipment в ~/ski.places/equiplist	16
4.20	Создать в домашнем каталоге файл abc1 и скопировать его в каталог ~/ski.places	16
4.21	Назвать его equiplist2	17
4.22	Каталог с именем equipment в каталоге ~/ski.places	17

4.23	~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment .	17
4.24	Создать и переместить	17
4.25	drwxr-r-...australia	18
4.26	drwx-x-x...play	18
4.27	-r-xr-r-...my_os	18
4.28	-rw-rw-r-...feathers	19
4.29	/etc/password	19
4.30	passwd	19
4.31	~/feathers в файл ~/file.old	20
4.32	~/file.old в каталог ~/play	20
4.33	~/play в каталог ~/fun	20
4.34	~/fun в каталог ~/play	21
4.35	Назвать его games	21
4.36	Лишить владельца файла ~/feathers права на чтение. Что произойдёт, если попытаться просмотреть файл ~/feathers командой cat? Что произойдёт, если попытаться скопировать файл ~/feathers? .	21
4.37	Дать владельцу файла ~/feathers право на чтение	22
4.38	Лишить владельца каталога ~/play права на выполнение. Перейти в каталог ~/play.	22
4.39	Дать владельцу каталога ~/play право на выполнение	22
4.40	man	23
4.41	man по mount	23
4.42	man по fsck	24
4.43	man по mkfs	24
4.44	man по kill	25
5.1	Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера	26

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 Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. 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.
3. В домашнем каталоге создайте директорию `~/ski.places`.
4. Переместите файл `equipment` в каталог `~/ski.places`.
5. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
6. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
7. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
8. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
9. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
10. 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.

11. Просмотрите содержимое файла `/etc/password`.
12. Скопируйте файл `~/feathers` в файл `~/file.old`.
13. Переместите файл `~/file.old` в каталог `~/play`.
14. Скопируйте каталог `~/play` в каталог `~/fun`.
15. Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.
16. 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.
17. Лишите владельца каталога `~/play` права на выполнение.
18. Перейдите в каталог `~/play`. Что произошло?
19. Дайте владельцу каталога `~/play` право на выполнение.
20. Прочитайте `man` по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

3 Теоретическое введение

Для **создания текстового файла** можно использовать команду `touch`. Формат команды:

- `touch имя-файла`

Для **просмотра файлов небольшого размера** можно использовать команду `cat`. Формат команды:

- `cat имя-файла`

Для **просмотра файлов** постранично удобнее использовать команду `less`. Формат команды:

- `less имя-файла`

Команда `cp` используется для копирования файлов и каталогов. Формат команды:

- `cp [-опции] исходный_файл целевой_файл`

Команды **`mv` и `mvdir`** предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv`:

- `mv [-опции] старый_файл новый_файл (рис. 3.1).`

Права доступа			
Право	Обозначение	Файл	Каталог
Чтение	r	Разрешены просмотр и копирование	Разрешён просмотр списка входящих файлов
Запись	w	Разрешены изменение и переименование	Разрешены создание и удаление файлов
Выполнение	x	Разрешено выполнение файла (скриптов и/или программ)	Разрешён доступ в каталог и есть возможность сделать его текущим

Рис. 3.1: Права доступа

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

- `chmod режим имя_файла`

Файловая система в Linux состоит из фалов 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).

4 Выполнение лабораторной работы

1. Выполнить все примеры, приведённые в первой части описания лабораторной работы (рис. 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12, 4.13, 4.14, 4.15).

```
[tsganina@fedora ~]$ cd
[tsganina@fedora ~]$ touch abc1
[tsganina@fedora ~]$ cp abc1 april
[tsganina@fedora ~]$ cp abc1 may
[tsganina@fedora ~]$ ls
abc1      work      Документы  Музыка
april     'Без названия.ipynb'  Загрузки  Общедоступные
bin       'Библиотека calibre'  Изображения  'Рабочий стол'
may       Видео     КомпАл     Шаблоны
```

Рис. 4.1: Скопировать файл ~/abc1 в файл april и в файл may

```
[tsganina@fedora ~]$ mkdir monthly
[tsganina@fedora ~]$ cp april may monthly
[tsganina@fedora ~]$ ls
abc1      monthly  Видео     КомпАл     Шаблоны
april     work     Документы Музыка
bin       'Без названия.ipynb'  Загрузки  Общедоступные
may       'Библиотека calibre'  Изображения  'Рабочий стол'
[tsganina@fedora ~]$ ls monthly
april may
```

Рис. 4.2: Скопировать файлы april и may в каталог monthly

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

Рис. 4.3: Скопировать файл monthly/may в файл с именем june, скопировать каталог monthly в каталог monthly.00, скопировать каталог monthly.00 в каталог /tmp

```
[tsganina@fedora ~]$ cd
[tsganina@fedora ~]$ mv april july
[tsganina@fedora ~]$ ls
abcl may work Видео Изображения Общедоступные
bin monthly 'Без названия.ipynb' Документы КомпАл 'Рабочий стол'
july monthly.00 'Библиотека calibre' Загрузки Музыка Шаблоны
```

Рис. 4.4: Изменить название файла april на july в домашнем каталоге

```
[tsganina@fedora ~]$ mv july monthly.00
[tsganina@fedora ~]$ ls
abcl monthly.00 Видео КомпАл Шаблоны
bin work Документы Музыка
may 'Без названия.ipynb' Загрузки Общедоступные
monthly 'Библиотека calibre' Изображения 'Рабочий стол'
[tsganina@fedora ~]$ ls monthly.00
july monthly
[tsganina@fedora ~]$ ls monthly.00/monthly
april june may
[tsganina@fedora ~]$ mv monthly.00 monthly.01
[tsganina@fedora ~]$ ls
abcl monthly.01 Видео КомпАл Шаблоны
bin work Документы Музыка
may 'Без названия.ipynb' Загрузки Общедоступные
monthly 'Библиотека calibre' Изображения 'Рабочий стол'
```

Рис. 4.5: Переместить файл july в каталог monthly.00, переименовать каталог monthly.00 в monthly.01

```
[tsganina@fedora ~]$ mkdir reports
[tsganina@fedora ~]$ mv monthly.01 reports
[tsganina@fedora ~]$ ls reports
monthly.01
[tsganina@fedora ~]$ mv reports/monthly.01 reports/monthly
[tsganina@fedora ~]$ ls reports
monthly
```

Рис. 4.6: Переместить каталог monthly.01 в каталог reports, переименовать каталог reports/monthly.01 в reports/monthly

```
[tsganina@fedora ~]$ cd
[tsganina@fedora ~]$ touch may
[tsganina@fedora ~]$ ls -l may
-rw-r--r--. 1 tsganina tsganina 0 map 7 15:29 may
[tsganina@fedora ~]$ chmod u+x may
[tsganina@fedora ~]$ ls -l may
-rwxr--r--. 1 tsganina tsganina 0 map 7 15:29 may
```

Рис. 4.7: Требуется создать файл ~/may с правом выполнения для владельца

```
[tsganina@fedora ~]$ chmod u-x may
[tsganina@fedora ~]$ ls -l may
-rw-r--r--. 1 tsganina tsganina 0 map 7 15:29 may
```

Рис. 4.8: Требуется лишить владельца файла ~/may права на выполнение

```
[tsganina@fedora ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[tsganina@fedora ~]$ ls
abcl  reports  Видео  КомпАл  Шаблоны
bin   work     Документы  Музыка
may   'Без названия.ipynb'  Загрузки  Общедоступные
monthly 'Библиотека calibre'  Изображения  'Рабочий стол'
```

Рис. 4.9: Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей

```
[tsganina@fedora ~]$ ls -ld monthly
d-wx--x--x. 1 tsganina tsganina 0 map 7 15:33 monthly
[tsganina@fedora ~]$ ls -ld monthly
d-wx--x--x. 1 tsganina tsganina 0 map 7 15:33 monthly
[tsganina@fedora ~]$
```

Рис. 4.10: Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей

```
[tsganina@fedora ~]$ chmod g+w abcl
[tsganina@fedora ~]$ ls -l abcl
-rw-rw-r--. 1 tsganina tsganina 0 map 7 15:17 abcl
```

Рис. 4.11: Требуется создать файл ~/abcl с правом записи для членов группы

```
[tsganina@fedora ~]$ mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=1048576,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=785292k,nr_inodes=819200,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursiveprot)
```

Рис. 4.12: Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой mount без параметров

```
[tsganina@fedora ~]$ cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Thu Feb 16 08:13:05 2023
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=0ef69679-b383-4b11-9e36-34199a2d3c63 / btrfs subvol=ro
ot,compress=zstd:1 0 0
UUID=37c0d827-f468-4587-acf6-b0e08614d3dd /boot ext4 defaults
1 2
UUID=9288-ED0C /boot/efi vfat umask=0077,shortname=winnt
0 2
UUID=0ef69679-b383-4b11-9e36-34199a2d3c63 /home btrfs subvol=home,compress=zstd:1 0 0
[tsganina@fedora ~]$
```

Рис. 4.13: Другой способ определения смонтированных в операционной системе файловых систем — просмотр файла/etc/fstab

```

[tsganina@fedora ~]$ df
Файловая система 1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs           4096             0      4096             0% /dev
tmpfs              1963228         0    1963228             0% /dev/shm
tmpfs              785292         1836    783456             1% /run
/dev/sda3          486722560      20798036  465511052           5% /
tmpfs              1963232         60    1963172             1% /tmp
/dev/sda3          486722560      20798036  465511052           5% /home
/dev/sda2          996780         273512    654456            30% /boot
/dev/sda1          613160         17796    595364             3% /boot/efi
tmpfs              392644         184    392460             1% /run/user/1000

```

Рис. 4.14: Для определения объёма свободного пространства на файловой системе можно воспользоваться командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования

```

[tsganina@fedora ~]$ fsck /dev/sda1
fsck из util-linux 2.38.1
fsck.fat 4.2 (2021-01-31)
open: Permission denied

```

Рис. 4.15: С помощью команды `fsck` можно проверить (а в ряде случаев восстановить) целостность файловой системы

2. Скопировать файл `/usr/include/sys/io.h` в домашний каталог и назвать его `equipment` (рис. 4.16).

```

[tsganina@fedora ~]$ pwd
/home/tsganina
[tsganina@fedora ~]$ cp /usr/include/sys/io.h /home/tsganina/equipment
[tsganina@fedora ~]$ ls
abcl      reports      Документы      Общедоступные
bin       work         Загрузки       'Рабочий стол'
equipment 'Без названия.ipynb'  Изображения    Шаблоны
may       'Библиотека calibre'  КомпАл
monthly   Видео         Музыка

```

Рис. 4.16: Копирование файла

3. В домашнем каталоге создать директорию `~/ski.plases` (рис. 4.17).

```
[tsganina@fedora ~]$ mkdir ~/ski.plases
[tsganina@fedora ~]$ ls | grep "ski"
ski.plases
[tsganina@fedora ~]$ ls -F | grep "ski"
ski.plases/
```

Рис. 4.17: ~/ski.plases

4. Переместить файл equipment в каталог ~/ski.plases (рис. 4.18).

```
[tsganina@fedora ~]$ mv equipment ~/ski.plases
[tsganina@fedora ~]$ ls ski.plases
equipment
```

Рис. 4.18: Перемещаю файл equipment в каталог ~/ski.plases

5. Переименовать файл ~/ski.plases/equipment в ~/ski.plases/equiplist (рис. 4.19).

```
[tsganina@fedora ~]$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
[tsganina@fedora ~]$ ls ski.plases
equiplist
```

Рис. 4.19: ~/ski.plases/equipment в ~/ski.plases/equiplist

6. Создать в домашнем каталоге файл abc1 и скопировать его в каталог ~/ski.plases, назвать его equiplist2 (рис. 4.20, 4.21).

```
[tsganina@fedora ~]$ touch abc1
[tsganina@fedora ~]$ ls
abc1  monthly  work      Видео      Изображения  Общедоступные
bin   reports  'Без названия.ipynb'  Документы  КомпАл      'Рабочий стол'
may   ski.plases  'Библиотека calibre'  Загрузки   Музыка      Шаблоны
[tsganina@fedora ~]$ cp abc1 ~/ski.plases
[tsganina@fedora ~]$ ls ski.plases
abc1 equiplist2
```

Рис. 4.20: Создать в домашнем каталоге файл abc1 и скопировать его в каталог ~/ski.plases


```
[tsganina@fedora ~]$ mv ~/ski.places/abc1 ~/ski.places/equiplist2
[tsganina@fedora ~]$ ls ski.places
equiplist  equiplist2
```

Рис. 4.21: Назвать его equiplist2

7. Создать каталог с именем equipment в каталоге ~/ski.places (рис. 4.22).

```
[tsganina@fedora ~]$ mkdir ~/ski.places/equipment
[tsganina@fedora ~]$ ls ski.places
equiplist  equiplist2  equipment
```

Рис. 4.22: Каталог с именем equipment в каталоге ~/ski.places

8. Переместить файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment (рис. 4.23).

```
[tsganina@fedora ~]$ mv ~/ski.places/equiplist ~/ski.places/equipment; mv ~/ski.places/equiplist2 ~/ski.places/equipment
[tsganina@fedora ~]$ ls ski.places
equipment
[tsganina@fedora ~]$ ls ski.places/equipment/
equiplist  equiplist2
```

Рис. 4.23: ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment

9. Создать и переместить каталог ~/newdir в каталог ~/ski.places и назвать его plans (рис. 4.24).

```
[tsganina@fedora ~]$ mkdir ~/newdir
[tsganina@fedora ~]$ ls
abc1  newdir  'Без названия.ipynb'  Загрузки  Общедоступные
bin   reports  'Библиотека calibre'  Изображения  'Рабочий стол'
may   ski.places  Видео  КомпАл  Шаблоны
monthly  work  Документы  Музыка
[tsganina@fedora ~]$ mv newdir ~/ski.places/plans
[tsganina@fedora ~]$ ls
abc1  monthly  work
bin   reports  'Без названия.ipynb'  Видео  Изображения  Общедоступные
may   ski.places  'Библиотека calibre'  Документы  КомпАл  'Рабочий стол'
[tsganina@fedora ~]$ ls ski.places
equipment  plans
```

Рис. 4.24: Создать и переместить

10. Определить 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 создать нужные файлы (рис. 4.25, 4.26, 4.27, 4.28).

```
[tsganina@fedora ~]$ mkdir australia
[tsganina@fedora ~]$ ls -ld australia
drwxr-xr-x. 1 tsganina tsganina 0 map  7 19:19 australia
[tsganina@fedora ~]$ chmod 744 australia
[tsganina@fedora ~]$ ls -ld australia
drwxr--r--. 1 tsganina tsganina 0 map  7 19:19 australia
```

Рис. 4.25: `drwxr-r-...australia`

```
[tsganina@fedora ~]$ mkdir play
[tsganina@fedora ~]$ ls -ld play
drwxr-xr-x. 1 tsganina tsganina 0 map  7 19:20 play
[tsganina@fedora ~]$
[tsganina@fedora ~]$ chmod 711 play
[tsganina@fedora ~]$ ls -ld play
drwx--x--x. 1 tsganina tsganina 0 map  7 19:20 play
```

Рис. 4.26: `drwx-x-x...play`

```
[tsganina@fedora ~]$ touch my_os
[tsganina@fedora ~]$ ls -ld my_os
-rw-r--r--. 1 tsganina tsganina 0 map  7 19:22 my_os
[tsganina@fedora ~]$ chmod 454 my_os
[tsganina@fedora ~]$ ls -ld my_os
-r--r-xr--. 1 tsganina tsganina 0 map  7 19:22 my_os
[tsganina@fedora ~]$ chmod 544 my_os
[tsganina@fedora ~]$ ls -ld my_os
-r-xr--r--. 1 tsganina tsganina 0 map  7 19:22 my_os
```

Рис. 4.27: `-r-xr-r-...my_os`

```
[tsganina@fedora ~]$ touch feathers
[tsganina@fedora ~]$ chmod 664
chmod: пропущен операнд после «664»
По команде «chmod --help» можно получить дополнительную информацию.
[tsganina@fedora ~]$ chmod 664 feathers
[tsganina@fedora ~]$ ls -ld feathers
-rw-rw-r--. 1 tsganina tsganina 0 map 7 19:24 feathers
```

Рис. 4.28: -rw-rw-r--...feathers

11. Просмотреть содержимое файла /etc/passwd (рис. 4.29, 4.30).

```
[tsganina@fedora ~]$ cat /etc/passwd
cat: /etc/passwd: Нет такого файла или каталога
[tsganina@fedora ~]$ ls /etc
abrt                               host.conf                         pm
adjtime                           hostname                        polkit-1
aliases                           hosts                            popt.d
alsa                               hp                               ppp
alternatives                      httpd                           printer
```

Рис. 4.29: /etc/passwd

```
[tsganina@fedora ~]$ ls /etc | grep "passwd"
passwd
passwd-
passwdqc.conf
[tsganina@fedora ~]$ ls -ld /etc/passwd
-rw-r--r--. 1 root root 2818 фев 19 16:34 /etc/passwd
[tsganina@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:/:usr/sbin/nologin
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
```

Рис. 4.30: passwd

12. Скопировать файл ~/feathers в файл ~/file.old (рис. 4.31).

```
[tsganina@fedora ~]$ cp ~/feathers ~/file.old
[tsganina@fedora ~]$ ls
abcl      may      ski.plases  Документы  Общедоступные
australia monthly  work        Загрузки   'Рабочий стол'
bin       my_os    'Без названия.ipynb'  Изображения  Шаблоны
feathers   play     'Библиотека calibre'  КомпАл
file.old  reports  Видео        Музыка
```

Рис. 4.31: ~/feathers в файл ~/file.old

13. Переместить файл ~/file.old в каталог ~/play (рис. 4.32).

```
[tsganina@fedora ~]$ mv file.old play
[tsganina@fedora ~]$ ls
abcl      monthly  work        Загрузки   'Рабочий стол'
australia my_os    'Без названия.ipynb'  Изображения  Шаблоны
bin       play     'Библиотека calibre'  КомпАл
feathers   reports  Видео        Музыка
may       ski.plases  Документы  Общедоступные
```

Рис. 4.32: ~/file.old в каталог ~/play

14. Скопировать каталог ~/play в каталог ~/fun (рис. 4.33).

```
[tsganina@fedora ~]$ cp play fun
cp: не указан -r; пропускается каталог 'play'
[tsganina@fedora ~]$ ls
abcl      monthly  work        Загрузки   'Рабочий стол'
australia my_os    'Без названия.ipynb'  Изображения  Шаблоны
bin       play     'Библиотека calibre'  КомпАл
feathers   reports  Видео        Музыка
may       ski.plases  Документы  Общедоступные
```

Рис. 4.33: ~/play в каталог ~/fun

15. Переместить каталог ~/fun в каталог ~/play и назвать его games (рис. 4.34, 4.35).

```
[tsganina@fedora ~]$ ls
abcl1    may    ski.plases    Документы    Общедоступные
australia  monthly  work    Загрузки    'Рабочий стол'
bin      my_os    'Без названия.ipynb'  Изображения  Шаблоны
feathers  play    'Библиотека calibre'  КомпАл
fun      reports  Видео    Музыка

[tsganina@fedora ~]$ ls fun
file.old

[tsganina@fedora ~]$ mv fun play
[tsganina@fedora ~]$ ls
abcl1    monthly  work    Загрузки    'Рабочий стол'
australia  my_os    'Без названия.ipynb'  Изображения  Шаблоны
bin      play    'Библиотека calibre'  КомпАл
feathers  reports  Видео    Музыка
may      ski.plases  Документы    Общедоступные
```

Рис. 4.34: ~/fun в каталог ~/play

```
[tsganina@fedora ~]$ cd play
[tsganina@fedora play]$ mv fun games
[tsganina@fedora play]$ ls
file.old  games
```

Рис. 4.35: Назвать его games

16. Лишить владельца файла ~/feathers права на чтение. Что произойдёт, если попытаться просмотреть файл ~/feathers командой cat? Что произойдёт, если попытаться скопировать файл ~/feathers? Дать владельцу файла ~/feathers право на чтение (рис. 4.36, 4.37).

```
[tsganina@fedora ~]$ chmod u-r feathers
[tsganina@fedora ~]$ ls -ld feathers
--w-rw-r--. 1 tsganina tsganina 0 map  7 19:24 feathers
[tsganina@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[tsganina@fedora ~]$ cp feathers
cp: после 'feathers' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
[tsganina@fedora ~]$ cp feathers feathers
cp: 'feathers' и 'feathers' - один и тот же файл
[tsganina@fedora ~]$ cp feathers feathers02
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[tsganina@fedora ~]$
```

Рис. 4.36: Лишить владельца файла ~/feathers права на чтение. Что произойдёт, если попытаться просмотреть файл ~/feathers командой cat? Что произойдёт, если попытаться скопировать файл ~/feathers?

После лишения владельца прав на чтение файла не получится ни просмотреть его, ни скопировать.

```
[tsganina@fedora ~]$ mkdir tmp05_lab
[tsganina@fedora ~]$ cp feathers ~/tmp05_lab
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
cp: не указан -r; пропускается каталог '/home/tsganina/'
[tsganina@fedora ~]$ cp feathers ~/tmp05_lab
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[tsganina@fedora ~]$ chmod u+r feathers
[tsganina@fedora ~]$ ls -ld feathers
-rw-rw-r--. 1 tsganina tsganina 0 map 7 19:24 feathers
```

Рис. 4.37: Дать владельцу файла ~/feathers право на чтение

17. Лишить владельца каталога ~/play права на выполнение. Перейти в каталог ~/play. Что произошло? Дать владельцу каталога ~/play право на выполнение (рис. 4.38, 4.39).

```
[tsganina@fedora ~]$ chmod u-x play
[tsganina@fedora ~]$ cd play
bash: cd: play: Отказано в доступе
[tsganina@fedora ~]$
```

Рис. 4.38: Лишить владельца каталога ~/play права на выполнение. Перейти в каталог ~/play.

Если лишить владельца каталога права на выполнение, то не получится больше перейти в этот каталог.

```
[tsganina@fedora ~]$ chmod u+x play
[tsganina@fedora ~]$ cd play
[tsganina@fedora play]$ cd ..
[tsganina@fedora ~]$
```

Рис. 4.39: Дать владельцу каталога ~/play право на выполнение

20. Прочитать man по командам mount, fsck, mkfs, kill и кратко их охарактеризовать, приведя примеры (рис. 4.40, 4.41, 4.42, 4.43, 4.44).

```

[tsganina@fedora ~]$ man mount
[tsganina@fedora ~]$ man fsck
[tsganina@fedora ~]$ man mkfs
[tsganina@fedora ~]$ man kill
[tsganina@fedora ~]$ man term
[tsganina@fedora ~]$

```

Рис. 4.40: man

```

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] devicemountpoint
                                     |
    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

```

Рис. 4.41: man по mount

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
1	Filesystem errors corrected
2	System should be rebooted
4	Filesystem errors left uncorrected

Рис. 4.42: man по fsck

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. 4.43: man по mkfs

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. 4.44: man по kill

5 Контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу (рис. 5.1).

```
[tsganina@fedora ~]$ findmnt -l
```

TARGET	SOURCE	FSTYPE	OPTIONS
/proc	proc	proc	rw,nosuid,nodev,noexec,relatime
/sys	sysfs	sysfs	rw,nosuid,nodev,noexec,relatime,se
/dev	devtmpfs	devtmpfs	rw,nosuid,seclabel,size=4096k,nr_i
/sys/kernel/security	securityfs	securityfs	rw,nosuid,nodev,noexec,relatime
/dev/shm	tmpfs	tmpfs	rw,nosuid,nodev,seclabel,inode64
/dev/pts	devpts	devpts	rw,nosuid,noexec,relatime,seclabel
/run	tmpfs	tmpfs	rw,nosuid,nodev,seclabel,size=7852
/sys/fs/cgroup	cgroup2	cgroup2	rw,nosuid,nodev,noexec,relatime,se
/sys/fs/pstore	pstore	pstore	rw,nosuid,nodev,noexec,relatime,se
/sys/firmware/efi/efivars	efivarfs	efivarfs	rw,nosuid,nodev,noexec,relatime
/sys/fs/bpf	bpf	bpf	rw,nosuid,nodev,noexec,relatime,mo
/	/dev/sda3[/root]		
		btrfs	rw,relatime,seclabel,compress=zstd
/sys/fs/selinux	selinuxfs	selinux	rw,nosuid,noexec,relatime
/proc/sys/fs/binfmt_misc	systemd-1	autofs	rw,relatime,fd=35,pgrp=1,timeout=0
/sys/kernel/tracing	tracefs	tracefs	rw,nosuid,nodev,noexec,relatime,se
/sys/kernel/debug	debugfs	debugfs	rw,nosuid,nodev,noexec,relatime,se
/dev/hugepages	hugetlbfs	hugetlb	rw,relatime,seclabel,pagesize=2M
/dev/mqueue	mqueue	mqueue	rw,nosuid,nodev,noexec,relatime,se
/sys/fs/fuse/connections	fusectl	fusectl	rw,nosuid,nodev,noexec,relatime
/sys/kernel/config	configfs	configfs	rw,nosuid,nodev,noexec,relatime
/tmp	tmpfs	tmpfs	rw,nosuid,nodev,seclabel,size=1963
/home	/dev/sda3[/home]		
		btrfs	rw,relatime,seclabel,compress=zstd
/boot	/dev/sda2	ext4	rw,relatime,seclabel
/boot/efi	/dev/sda1	vfat	rw,relatime,fmask=0077,dmask=0077,
/proc/sys/fs/binfmt_misc	binfmt_misc	binfmt_	rw,nosuid,nodev,noexec,relatime
/var/lib/nfs/rpc_pipefs	sunrpc	rpc_pipefs	rw,relatime
/run/user/1000	tmpfs	tmpfs	rw,nosuid,nodev,relatime,seclabel,
/run/user/1000/gvfs	gvfsd-fuse	fuse.gv	rw,nosuid,nodev,relatime,user_id=1
/run/user/1000/doc	portal	fuse.po	rw,nosuid,nodev,relatime,user_id=1
/tmp/.mount_jetbraluUEwF	jetbrains-toolbox		
		fuse.je	ro,nosuid,nodev,relatime,user_id=1

Рис. 5.1: Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера

- `proc`. Файловая система `proc` является важным источником информации о вашей Linux-системе, который попросту нельзя игнорировать. Вообще, `proc`

является псевдо- или виртуальной файловой системой, которая предоставляет пользователям доступ к внутренним структурам ядра Linux. Другими словами, `proc` не является реальной файловой системой в обычном смысле; она располагается исключительно в оперативной памяти, а не на диске. При этом она автоматически монтируется системой.

- `Sysfs` - отправляет данные в пространство пользователя с помощью виртуальных файлов. Эти данные содержат данные о различных подсистемах ядра, аппаратных устройствах и связанных с ними драйверах устройств.
- `tmpfs` и `devtmpfs` - они относятся к энергозависимой памяти.
- `devpts` - обеспечивает доступ к терминалам `pseudo` (PTY).
- `cgroup2` - Неверно ведущий себя процесс может создавать тонны процессов через ветвления, запуская некую бомбу ветвлений и сокрушая своё ядро. Это означает, что нам требуется ввести некий способ контроля ресурсов для процессов в пределах заданного пространства имён. Это достигается через механизм, носящий название групп контроля (`control groups`), обычно именуемых `cgroups`. `cgroups` работают под понятием контроллеров `cgroup` и представляются в файловой системе с названием `cgroupfs` в самом ядре Linux. В настоящее время применяется `cgroup v2` версия `cgroups`.
- `pstore` - был введен в Linux для записи информации (например, `dmesg tail`) при выключении. `Pstore` не зависит от `kdump` и может запускаться до него. В определенных сценариях (например, хосты/гости с корневыми файловыми системами на NFS/iSCSI, где произошел сбой сетевого программного и/или аппаратного обеспечения), `pstore` может содержать информацию, доступную для посмертной отладки, которая не может быть получена иным образом.
- `brf` - это псевдо-файловая система, существующая только в памяти, которая позволяет создавать файлы, ссылающиеся на объекты BPF.

- `btrfs` - файловая система для Linux, основанная на структурах B-деревьев и работающая по принципу «копирование при записи» (copy-on-write). Опубликована корпорацией Oracle в 2007 году под лицензией GNU General Public License.
- `selinux` - Как и файловая система `/proc`, `/selinux` является псевдофайловой системой. Новая реализация SE Linux использует расширенные атрибуты для хранения контекста безопасности.
- `tracefs` - файловая система для задач трассировки Linux
- `debugfs` - DebugFS является самой известной утилитой, предназначенной для работы с файловыми системами EXT2FS и EXT3FS.
- `hugetlb` - использует страницы большого размера, что позволяет кэшировать больше адресов за раз.
- `mqueue` - обеспечивает необходимую поддержку ядра для библиотеки пользовательского пространства, которая реализует интерфейсы очереди сообщений POSIX.
- `fusectl` - это простой интерфейс для программ пользовательского пространства для экспорта виртуальной файловой системы в ядро Linux.
- `configfs` - Виртуальная файловая система, представляющая состояние ядра операционной системы и запущенных процессов в виде файлов.
- `ext4` - журналируемая файловая система, используемая преимущественно в операционных системах с ядром Linux, созданная на базе `ext3` в 2006 году.
- `fuse.gv`, `fuse.po`, `fuse.je` - FUSE (файловая система в пользовательском пространстве) — это интерфейс для программ пользовательского пространства для экспорта файловой системы в ядро 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:

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

5. Как создаётся файловая система?

- `mkfs` - позволяет создать файловую систему Linux.

Создать файловую систему linux, семейства ext, на устройстве можно с помощью команды `mkfs`. Ее синтаксис выглядит следующим образом:

```
sudo mkfs -t тип устройства
```

Доступны дополнительные параметры:

- `c` - проверить устройство на наличие битых секторов
- `b` - размер блока файловой системы
- `j` - использовать журналирование для ext3

- L - задать метку раздела
- v - показать подробную информацию о процессе работы
- V - версия программы

6. Дайте характеристику командам для просмотра текстовых файлов.

- `cat <имя_файла>` - Это самая простая и, пожалуй, самая популярная команда для просмотра файла в Linux. Cat просто печатает содержимое файла на стандартном экране, т.е. на экране. В основном используется для небольших файлов.
- `Less <имя_файла>`. Команда Less просматривает файл по одной странице за раз.
- `Head <имя_файла>`. Команда Head — это еще один способ просмотра текстового файла, но с небольшой разницей. Команда head отображает первые 10 строк текстового файла по умолчанию. Вы можете изменить это поведение, используя опции с командой head, но основной принцип остается тем же: команда head начинает работать с заголовка (начала) файла.
- `Tail <имя_файла>`. Команда Tail в Linux аналогична и все же противоположна команде head. В то время как команда head отображает файл с начала, команда tail отображает файл с конца. По умолчанию команда tail отображает последние 10 строк файла. Команды Head и Tail могут быть объединены для отображения выбранных строк из файла. Вы также можете использовать команду tail для просмотра изменений, внесенных в файл в режиме реального времени.

7. Приведите основные возможности команды cp в Linux.

Это сокращение от сору, и она делает именно то, что предполагает ее название: она копирует. cp используется для копирования файлов из одного местоположения в другое. cp также можно использовать для копирования всех каталогов

в новое место. Можно использовать эту команду для копирования нескольких файлов и каталогов.

8. Приведите основные возможности команды mv в Linux.

Команда mv используется для перемещения файлов из одного каталога в другой. Также команда mv используется для переименования файла в системах Linux.

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

Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенными файлами и каталогами.

Каждый файл можно изменять по трём параметрам доступа. Вот они:

- Чтение - разрешает получать содержимое файла, но на запись нет. Для каталога позволяет получить список файлов и каталогов, расположенных в нем;
- Запись - разрешает записывать новые данные в файл или изменять существующие, а также позволяет создавать и изменять файлы и каталоги;
- Выполнение - вы не можете выполнить программу, если у нее нет флага выполнения. Этот атрибут устанавливается для всех программ и скриптов, именно с помощью него система может понять, что этот файл нужно запускать как программу.

Чтобы получить доступ к файлам в Linux, используются разрешения. Эти разрешения назначаются трем объектам: файлу, группе и другому объекту (то есть всем остальным). Изменить права доступа можно при помощи команды **chmod**:

`chmod <параметры изменения> <имя_файла/каталога>`

6 Выводы

В ходе выполнения этой лабораторной 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.

Список литературы

1. Руководство к выполнению лабораторной работы №5
2. Статья “Chmod (777, 755, 444) — настройка прав доступа к файлам и папкам”