

Отчет по лабораторной работе №7

Дисциплина: Операционные системы

Иванов Сергей Владимирович

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 4 |
| 2 | Задание | 5 |
| 3 | Выполнение лабораторной работы | 6 |
| 4 | Контрольные вопросы | 13 |
| 5 | Выводы | 17 |

Список иллюстраций

| | | |
|------|-------------------------------------|----|
| 3.1 | Создание файла | 6 |
| 3.2 | Создание директории | 6 |
| 3.3 | Копирование файла | 6 |
| 3.4 | Создание директории | 7 |
| 3.5 | Переименовывание файла | 7 |
| 3.6 | Создание директории | 7 |
| 3.7 | Изменение прав доступа | 8 |
| 3.8 | Изменение прав доступа | 8 |
| 3.9 | Изменение прав доступа | 8 |
| 3.10 | Проверка файловой системы | 9 |
| 3.11 | Копирование файла | 9 |
| 3.12 | Создание файла | 9 |
| 3.13 | Создание директории | 10 |
| 3.14 | Изменение прав доступа | 10 |
| 3.15 | Изменение прав доступа | 10 |
| 3.16 | Изменение прав доступа | 10 |
| 3.17 | Чтение файла | 11 |
| 3.18 | Копирование файла | 11 |
| 3.19 | Изменение прав доступа | 11 |
| 3.20 | Работа с правами доступа | 12 |
| 3.21 | Чтение документации | 12 |

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, 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 Выполнение лабораторной работы

Создаем файл, дважды копируем его с новыми именами и проверяем, что все команды были выполнены корректно. (рис. 1).

```
[svivanov1@svivanov1 ~]$ touch abc1
[svivanov1@svivanov1 ~]$ cp abc1 april
[svivanov1@svivanov1 ~]$ cp abc1 may
[svivanov1@svivanov1 ~]$ ls
abc1  git-extended  presentation  work  Документы  Музыка  Шаблоны
april  may           report        work2  Загрузки  Общедоступные
bin   package.json  touch        Видео  Изображения  'Рабочий стол'
```

Рис. 3.1: Создание файла

Создаю директорию, копирую в нее два файла, созданных на прошлом этапе, проверяю, что все скопировалось (рис. 2).

```
[svivanov1@svivanov1 ~]$ mkdir monthly
[svivanov1@svivanov1 ~]$ cp april may monthly/
[svivanov1@svivanov1 ~]$ ls monthly/
april  may
```

Рис. 3.2: Создание директории

Копирую файл, находящийся не в текущей директории в файл с новым именем тоже не текущей директории (рис. 3).

```
[svivanov1@svivanov1 ~]$ cp monthly/may monthly/june
[svivanov1@svivanov1 ~]$ ls monthly
april  june  may
```

Рис. 3.3: 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. 4).

```
[svivanov1@svivanov1 ~]$ mkdir monthly.00
[svivanov1@svivanov1 ~]$ cp -r monthly /tmp
[svivanov1@svivanov1 ~]$ ls monthly
april  june  may
[svivanov1@svivanov1 ~]$ cp -r monthly monthly.00
[svivanov1@svivanov1 ~]$ ls monthly.00
monthly
```

Рис. 3.4: Создание директории

Переименовываю файл, затем перемещаю его в каталог (рис. 5).

```
[svivanov1@svivanov1 ~]$ mv april july
[svivanov1@svivanov1 ~]$ mv july monthly.00
[svivanov1@svivanov1 ~]$ ls monthly.00/
july  monthly
[svivanov1@svivanov1 ~]$
```

Рис. 3.5: Переименовывание файла

Создаю новую директорию, переименовываю monthly.00 в monthly.01, перемещаю директорию в директорию reports, переименовываю эту директорию, убираю из названия 01 (рис. 6).

```
[svivanov1@svivanov1 ~]$ mkdir reports
[svivanov1@svivanov1 ~]$ mv monthly.00 monthly.01
[svivanov1@svivanov1 ~]$ mv monthly.01/ reports/
[svivanov1@svivanov1 ~]$ mv reports/monthly.01 reports/monthly
[svivanov1@svivanov1 ~]$
```

Рис. 3.6: Создание директории

Создаю пустой файл, проверяю права доступа у него, изменяю права доступа, добавляя пользователю (создателю) возможность выполнять файл (рис. 7).

```
[svivanov1@svivanov1 ~]$ touch may
[svivanov1@svivanov1 ~]$ ls -l may
-rw-r--r--. 1 svivanov1 svivanov1 0 map 13 19:11 may
[svivanov1@svivanov1 ~]$ chmod u+x may
[svivanov1@svivanov1 ~]$ ls -l may
-rwxr--r--. 1 svivanov1 svivanov1 0 map 13 19:11 may
[svivanov1@svivanov1 ~]$ chmod u-x may
[svivanov1@svivanov1 ~]$ ls -l may
-rw-r--r--. 1 svivanov1 svivanov1 0 map 13 19:11 may
```

Рис. 3.7: Изменение прав доступа

Меняю права доступа у директории: группы и остальные пользователи не смогут ее прочесть (рис. 8).

```
[svivanov1@svivanov1 ~]$ chmod g-r monthly
[svivanov1@svivanov1 ~]$ chmod o-r monthly
[svivanov1@svivanov1 ~]$ ls -l monthly
итого 0
-rw-r--r--. 1 svivanov1 svivanov1 0 map 13 19:01 april
-rw-r--r--. 1 svivanov1 svivanov1 0 map 13 19:02 june
-rw-r--r--. 1 svivanov1 svivanov1 0 map 13 19:01 may
```

Рис. 3.8: Изменение прав доступа

Изменяю права доступа у директории, запрещаю группам и остальным пользователям читать. Создаю новый пустой файл, даю ему права доступа: группы могут читать и писать содержимое (рис. 9).

```
[svivanov1@svivanov1 ~]$ touch abc1
[svivanov1@svivanov1 ~]$ chmod g+w abc1
[svivanov1@svivanov1 ~]$ ls -l abc1
-rw-rw-r--. 1 svivanov1 svivanov1 0 map 13 19:17 abc1
[svivanov1@svivanov1 ~]$
```

Рис. 3.9: Изменение прав доступа

Проверяю файловую систему (рис. 10).


```
[svivanov1@svivanov1 ~]$ sudo fsck /dev/sda1
[sudo] пароль для svivanov1:
fsck from util-linux 2.39.3
fsck.fat 4.2 (2021-01-31)
There are differences between boot sector and its backup.
This is mostly harmless. Differences: (offset:original/backup)
 65:01/00
1) Copy original to backup
2) Copy backup to original
3) No action
[123?q]? 3
Dirty bit is set. Fs was not properly unmounted and some data may be corrupt.
1) Remove dirty bit
2) No action
[12?q]? 2
/dev/sda1: 24 files, 4873/153290 clusters
```

Рис. 3.10: Проверка файловой системы

Копирую файл в домашний каталог с новым именем, создаю новую пустую директорию, перемещаю файл в эту директорию, переименовываю файл (рис. 11).

```
[svivanov1@svivanov1 ~]$ cp /usr/include/sys/io.h equipment
[svivanov1@svivanov1 ~]$ ls
abc1      git-extended  package.json  reports  work2  Загрузки  Общедоступные
bin       may          presentation  touch    Видео  Изображения  'Рабочий стол'
equipment monthly      report        work     Документы  Музыка  Шаблоны
[svivanov1@svivanov1 ~]$ mkdir ski.places
[svivanov1@svivanov1 ~]$ mv equipment ski.places/equipment
[svivanov1@svivanov1 ~]$ ls
abc1      may          presentation  ski.places  work2  Загрузки  Общедоступные
bin       monthly      report        touch       Видео  Изображения  'Рабочий стол'
git-extended package.json  reports       work        Документы  Музыка  Шаблоны
[svivanov1@svivanov1 ~]$ ls ski.places/
equipment
[svivanov1@svivanov1 ~]$ mv ski.places/equipment ski.places/equiplist
[svivanov1@svivanov1 ~]$ ls ski.places/
equiplist
```

Рис. 3.11: 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. 12).

```
[svivanov1@svivanov1 ~]$ touch abc1
[svivanov1@svivanov1 ~]$ cp abc1 ski.places/equiplist2
[svivanov1@svivanov1 ~]$ mkdir ski.places/equipment
[svivanov1@svivanov1 ~]$ mv ski.places/equiplist ski.places/equiplist2 ski.places/equipment/
[svivanov1@svivanov1 ~]$ ls ski.places/equipment/
equiplist equiplist2
```

Рис. 3.12: Создание файла

Создаю новую директорию, в этой же строчке перемещаю ее с новым именем в директорию, созданную в прошлый раз (рис. 13).

```
[svivanov1@svivanov1 ~]$ mkdir newdir ; mv newdir ski.places/plans
[svivanov1@svivanov1 ~]$ ls ski.places/
equipment plans
```

Рис. 3.13: Создание директории

Проверяю, какие права нужно поменять и как, чтобы у новой директории были нужные по заданию права (рис. 14).

```
[svivanov1@svivanov1 ~]$ mkdir australia
[svivanov1@svivanov1 ~]$ chmod u+x australia
[svivanov1@svivanov1 ~]$ ls -l
итого 8
-rw-rw-r--. 1 svivanov1 svivanov1 0 map 13 19:29 abc1
drwxr-xr-x. 1 svivanov1 svivanov1 0 map 13 19:32 australia
```

Рис. 3.14: Изменение прав доступа

Проверяю, какие права нужно поменять и как, чтобы у новых файлов были нужные по заданию права (рис. 15).

```
[svivanov1@svivanov1 ~]$ mkdir play
[svivanov1@svivanov1 ~]$ chmod u+x play
[svivanov1@svivanov1 ~]$ chmod g-r+x play/
[svivanov1@svivanov1 ~]$ chmod o-r+x play/
[svivanov1@svivanov1 ~]$ ls -l
drwx--x--x. 1 svivanov1 svivanov1 0 map 13 19:42 play
```

Рис. 3.15: Изменение прав 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. 16).

```
[svivanov1@svivanov1 ~]$ touch my_os
[svivanov1@svivanov1 ~]$ chmod u+x-w my_os
[svivanov1@svivanov1 ~]$ ls -l my_os
-r-xr--r--. 1 svivanov1 svivanov1 0 map 13 19:48 my_os
[svivanov1@svivanov1 ~]$ touch feathers
[svivanov1@svivanov1 ~]$ chmod g+w feathers
[svivanov1@svivanov1 ~]$ ls -l feathers
-rw-rw-r--. 1 svivanov1 svivanov1 0 map 13 19:48 feathers
```

Рис. 3.16: Изменение прав доступа

Читаю содержимое файла (рис. 17).

```
[svivanov1@svivanov1 ~]$ cat /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
```

Рис. 3.17: 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. 18).

```
[svivanov1@svivanov1 ~]$ cp feathers file.old
[svivanov1@svivanov1 ~]$ mv file.old play/file.old
[svivanov1@svivanov1 ~]$ cp -r play/ fun/
[svivanov1@svivanov1 ~]$ cp -r fun/ play/games
[svivanov1@svivanov1 ~]$ ls play/
file.old  games
```

Рис. 3.18: 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. 19).

```
[svivanov1@svivanov1 ~]$ chmod u-r feathers
[svivanov1@svivanov1 ~]$ ls -l feathers
--w-rw-r--. 1 svivanov1 svivanov1 0 map 13 19:48 feathers
[svivanov1@svivanov1 ~]$ cat feathers
cat: feathers: Отказано в доступе
[svivanov1@svivanov1 ~]$ cp feathers feathers1
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[svivanov1@svivanov1 ~]$ chmod u+r feathers
```

Рис. 3.19: Изменение прав доступа

Убираю у директории право на исполнение для пользователя, пытаюсь в нее войти - отказано в доступе, возвращаю все права (рис. 20).

```
[svivanov1@svivanov1 ~]$ chmod u-x play/
[svivanov1@svivanov1 ~]$ cd play/
bash: cd: play/: Отказано в доступе
[svivanov1@svivanov1 ~]$ chmod u+x play/
[svivanov1@svivanov1 ~]$ cd play/
[svivanov1@svivanov1 play]$
```

Рис. 3.20: Работа с правами доступа

Я прочитал описание каждой из четырех команд с помощью `man` - `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. (рис. 21)

```
[svivanov1@svivanov1 ~]$ man mount
[svivanov1@svivanov1 ~]$ man fsck
[svivanov1@svivanov1 ~]$ man mkfs
[svivanov1@svivanov1 ~]$ man kill
[svivanov1@svivanov1 ~]$
```

Рис. 3.21: Чтение документации

4 Контрольные вопросы

1. На жестком диске компьютера, на котором проводилась лабораторная работа, были установлены различные файловые системы, такие как Ext2, Ext3, Ext4, Btrfs и Extended Filesystem. Файловая система Extended Filesystem является стандартной для Linux и отличается высокой стабильностью и большим количеством функций. Ext2, Ext3 и Ext4 представляют собой последовательные версии файловой системы Ext, с каждой последующей версией вносятся улучшения, включая стабильность и увеличение максимального размера раздела. Btrfs - новая файловая система, ориентированная на отказоустойчивость, легкость администрирования и восстановления данных, хотя некоторые пользователи считают её нестабильной.
2. Общая структура файловой системы включает корневой каталог “/”, который содержит всю иерархию системы, и другие директории первого уровня, такие как: /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. Для доступа к содержимому файловой системы операционная система должна выполнить операцию монтирования тома.
4. Основные причины нарушения целостности файловой системы могут включать отсутствие синхронизации между образом файловой системы в памяти и данными на диске при аварийном выключении, что может привести к различным ошибкам. Для исправления повреждений файловой системы можно использовать инструменты для восстановления данных или провести проверку и восстановление с помощью соответствующих командных инструментов.
5. Файловая система создается с помощью команды mkfs, которая позволяет создать файловую систему Linux.
6. Команда cat используется для вывода содержимого файла на стандартное устройство вывода. Команды head и tail позволяют просматривать первые и последние строки файла соответственно. Команда tac аналогична cat, но выводит строки в обратном порядке. Для просмотра больших текстовых

файлов часто используются команды `more` и `less` для постраничного просмотра.

7. Команда `cp` в Linux используется для копирования или перемещения файлов и директорий.
8. Команда `mv` в Linux используется для переименования или перемещения файлов и директорий.
9. Права доступа определяют, кто и как может обращаться к файлам или каталогам. Они могут быть изменены с помощью команды `chmod`, доступной владельцу файла или директории или пользователю с правами администратора.

5 Выводы

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