

# **Лабораторная работа No 5. Анализ файловой системы Linux. Команды для работы с файлами и каталогами**

**Операционные системы**

Норсоян Шушаник Гагиковна

# Содержание

1	Цель работы	5
2	Теоретическое введение	7
3	Выполнение лабораторной работы	8
4	Контрольные вопросы	15
	Список литературы	22

## Список иллюстраций

3.1	Выполняю примеры из лабораторной . . . . .	8
3.2	Продолжаю выполнять примеры из лабораторной . . . . .	9
3.3	Продолжаю выполнять примеры из лабораторной . . . . .	9
3.4	Продолжаю выполнять примеры из лабораторной . . . . .	10
3.5	fsck . . . . .	10
3.6	Выполняю действия записанные в методичке . . . . .	11
3.7	Создаю директорию ski.places . . . . .	11
3.8	меняю название файла . . . . .	11
3.9	Создаю новый каталог . . . . .	12
3.10	Создаю новый каталог . . . . .	12
3.11	Определяю опции команды chmod . . . . .	13
3.12	желаю всем счастья . . . . .	13
3.13	выполняю задания . . . . .	14

# Список таблиц

2.1	Описание некоторых каталогов файловой системы GNU Linux . .	7
-----	---	---

# 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. # Задание

2 Задание 1. Выполните все примеры, приведённые в первой части описания лабораторной работы. 2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения: 2.1. 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. 2.2. В домашнем каталоге создайте директорию ~/ski.places. 2.3. Переместите файл equipment в каталог ~/ski.places. 2.4. Переименуйте файл ~/ski.places/equipment в ~/ski.places/equiplist. 2.5. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог '~/ski.places', назовите его equiplist2'. 2.6. Создайте каталог с именем equipment в каталоге ~/ski.places. 2.7. Переместите файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment. 2.8. Создайте и переместите каталог '~/newdir' в каталог '~/ski.places' и назовите его plans. 3. Определите опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет: 3.1. drwxr-r- ... australia 3.2. drwx-x-x ... play 3.3. -r-xr-r- ... my\_os 3.4. -rw-rw-r- ... feathers При необходимости создайте нужные файлы. 4. Прделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды: 4.1. Просмотрите содержимое файла

/etc/passwd. 4.2. Скопируйте файл ~/feathers в файл ~/file.old. 4.3. Переместите файл ~/file.old в каталог ~/play. 4.4. Скопируйте каталог ~/play в каталог ~/fun. 4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games. 4.6. Лишите владельца файла ~/feathers права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat? 4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers? 4.9. Дайте владельцу файла ~/feathers право на чтение. 4.10. Лишите владельца каталога ~/play права на выполнение. 4.11. Перейдите в каталог ~/play. Что произошло? 4.12. Дайте владельцу каталога ~/play право на выполнение. 5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

## 2 Теоретическое введение

Здесь описываются теоретические аспекты, связанные с выполнением работы. Например, в табл. 2.1 приведено краткое описание стандартных каталогов Unix.


Таблица 2.1: Описание некоторых каталогов файловой системы GNU Linux

Имя каталога	Описание каталога
/	Корневая директория, содержащая всю файловую
/bin	Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям
/etc	Общесистемные конфигурационные файлы и файлы конфигурации установленных программ
/home	Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя
/media	Точки монтирования для сменных носителей
/root	Домашняя директория пользователя root
/tmp	Временные файлы
/usr	Вторичная иерархия для данных пользователя

Более подробно об Unix см. в [1–6].

### 3 Выполнение лабораторной работы

1. Выполнила все примеры, приведённые в первой части описания лабораторной работы. Скопировала файл ~/abc1 в файл april и в файл may. Скопировала файлы april и may в каталог monthly. Скопировала файл monthly/may в файл с именем june. Скопировала каталог monthly в каталог monthly.00. Скопировала каталог monthly.00 в каталог /tmp



```
sgnorsoyan@dk8n80 ~ $ cd
sgnorsoyan@dk8n80 ~ $ touch abc1
sgnorsoyan@dk8n80 ~ $ cp abc1 april
sgnorsoyan@dk8n80 ~ $ cp abc1 may
sgnorsoyan@dk8n80 ~ $ mkdir monthly
sgnorsoyan@dk8n80 ~ $ cp april may monthly
sgnorsoyan@dk8n80 ~ $ cp monthly/may monthly/june
sgnorsoyan@dk8n80 ~ $ ls monthly
april  june  may
sgnorsoyan@dk8n80 ~ $ mkdir monthly.00
sgnorsoyan@dk8n80 ~ $ cp -r monthly monthly.00
sgnorsoyan@dk8n80 ~ $ cp -r monthly.00 /tmp
sgnorsoyan@dk8n80 ~ $
```

Рис. 3.1: Выполняю примеры из лабораторной

2. Изменила название файла april на july в домашнем каталоге. Переместила файл july в каталог monthly.00. Переименовала каталог monthly.00 в monthly.01. Переместила каталог monthly.01 в каталог reports. Переименовала каталог reports/monthly.01 в reports/monthly



```
sgnorsoyan@dk8n80 ~ $ mv monthly.00 monthly.01
sgnorsoyan@dk8n80 ~ $ mkdir reports
sgnorsoyan@dk8n80 ~ $ mv monthly.01 reports
sgnorsoyan@dk8n80 ~ $ mv reports/monthly.01 reports/monthly
sgnorsoyan@dk8n80 ~ $
```

Рис. 3.2: Продолжаю выполнять примеры из лабораторной

3. Создала файл ~/may с правом выполнения для владельца. Лишила владельца файла ~/may права на выполнение. Создала каталог monthly с запретом на чтение для членов группы и всех остальных пользователей. Создала файл ~/abc1 с правом записи для членов группы.

```
sgnorsoyan@dk8n80 ~ $ cd
sgnorsoyan@dk8n80 ~ $ touch may
sgnorsoyan@dk8n80 ~ $ ls -l may
-rw-r--r-- 1 sgnorsoyan studsci 0 мар  9 13:53 may
sgnorsoyan@dk8n80 ~ $ chmod u+x may
sgnorsoyan@dk8n80 ~ $ ls -l may
-rwxr--r-- 1 sgnorsoyan studsci 0 мар  9 13:53 may
sgnorsoyan@dk8n80 ~ $ chmod u-x may
sgnorsoyan@dk8n80 ~ $ ls -l may
-rw-r--r-- 1 sgnorsoyan studsci 0 мар  9 13:53 may
sgnorsoyan@dk8n80 ~ $ cd
sgnorsoyan@dk8n80 ~ $ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
sgnorsoyan@dk8n80 ~ $ chmod g-r, o-r monthly
chmod: неверный режим: «g-r»
По команде «chmod --help» можно получить дополнительную информацию.
sgnorsoyan@dk8n80 ~ $ cd
sgnorsoyan@dk8n80 ~ $ chmod g-r, o-r monthly
chmod: неверный режим: «g-r»
По команде «chmod --help» можно получить дополнительную информацию.
sgnorsoyan@dk8n80 ~ $ help
GNU bash, версия 5.1.16(1)-release (x86_64-pc-linux-gnu)
Показание ниже команды определены внутри командного процессора. Чтобы вывести полный список, введите «help».
Чтобы вывести справку по функции «name», введите «help name».
Чтобы вывести справку по командному процессору, введите «info bash».
Чтобы вывести справку по командам, которые отсутствуют в этом списке, введите «man -k» или «info».

Звездочка (*) рядом с названием команды означает, что эта команда отключена.

задание [a]
(( [выражение] ))
_ файл [аргументы]
_
[ аргумент... ]
[[ [выражение] ]]
alias [-p] имя[=значение] ... ]
bg [задание ...]
bind [-lvsPSvx] [-m раскладка] [-f файл] [-q имя] [-u name] [-r послед_клавиш] [-x n]
break [n]
builtin [встр_команда [аргумент ...]]
caller [выражение]
case СЛОВА in [ШАБЛОН [ [ШАБЛОН]... ] КОМАНДЫ ;;]... esac
cd [-L[-P [-c]] [-o]] [каталог]
command [-pv] команда [аргумент ...]
compgen [-abdcfgjkquv] [-o option] [-A action] [-G globpat] [-W wordlist] [-f functi>
complete [-abdcfgjkquv] [-pr] [-DEI] [-o option] [-A action] [-G globpat] [-W wordli>
coport [-o]o option] [-DEI] [name ...]
continue [n]
coproc [ИМЯ] команда [перенаправления]
declare [-aAffgIlInrtux] [-p] [name[=value] ...]
```

Рис. 3.3: Продолжаю выполнять примеры из лабораторной

```
sgnorsoyan@dk8n80 ~ $ cd
sgnorsoyan@dk8n80 ~ $ touch abc1
sgnorsoyan@dk8n80 ~ $ chmod g+w abc1
sgnorsoyan@dk8n80 ~ $
```

Рис. 3.4: Продолжаю выполнять примеры из лабораторной

4. Воспользовалась командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования, для определения объёма свободного пространства на файловой системе. С помощью команды `fsck` проверила целостность файловой системы

```
sgnorsoyan@dk8n80 ~ $ fsck /dev/sda1
fsck из util-linux 2.38.1
e2fsck 1.46.5 (30-Dec-2021)
fsck.ext2: Отказано в доступе while trying to open /dev/sda1
You must have r/w access to the filesystem or be root
```

Рис. 3.5: fsck

5. Выполняю действия, записанные в методичке. Копирую файл `/usr/include/sys/io.h` в домашний каталог командой `cp` и называю его `equipment` при помощи команды `mv`. Проверяю наличие нужного нам файла в каталоге при помощи команды `ls`.

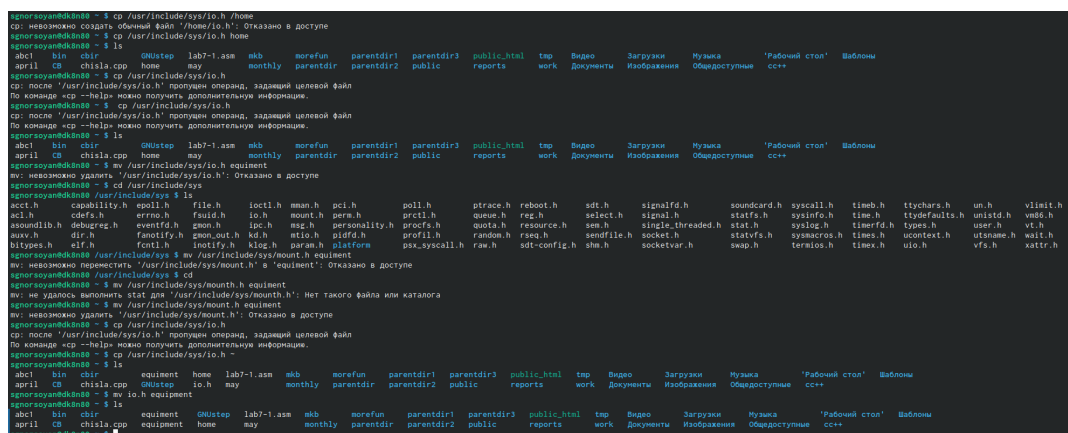


Рис. 3.6: Выполняю действия зписанные в методичке

6. Создаю директорию `ski.places` командой `mkdir` в домашнем каталоге и перемещаю файл `equipment` туда. Проверяю наличие всего командой `ls`. Так же меняю название на `equiplist`.



Рис. 3.7: Создаю директорию `ski.places`

7. Файл `abc1` был уже создан, копирую его в каталог `ski.places` и называю новым именем `equiplist2`. Проверяю всё через `ls`.



Рис. 3.8: меняю название файла

8. Создаю новый каталог в каталоге ski.places под названием equipment. Перемещаю файлы ski.places/equiplist и equiplist2 в каталог ski.places/equipment командой mv

```
sgnorsoyan@dk8n80 ~/ski.places $ cd
sgnorsoyan@dk8n80 ~ $ cd ski.places
sgnorsoyan@dk8n80 ~/ski.places $ mkdir equipment
sgnorsoyan@dk8n80 ~/ski.places $ ls
equiplist  equiplist2  equipment
sgnorsoyan@dk8n80 ~/ski.places $ mv equiplist equiplist2 equipment
sgnorsoyan@dk8n80 ~/ski.places $ cd equipment
sgnorsoyan@dk8n80 ~/ski.places/equipment $ ls
equiplist  equiplist2
sgnorsoyan@dk8n80 ~/ski.places/equipment $
```

Рис. 3.9: Создаю новый каталог

9. Создаю в каталоге ski.places новый каталог newdir и называю его plans

```
sgnorsoyan@dk8n80 ~ $ rmdir newdir
rmdir: не удалось удалить 'newdir': Нет такого файла или каталога
sgnorsoyan@dk8n80 ~ $ mkdir newdir
sgnorsoyan@dk8n80 ~ $ cp newdir ski.places
cp: не указан -r; пропускается каталог 'newdir'
sgnorsoyan@dk8n80 ~ $ cp -r newdir ski.places
sgnorsoyan@dk8n80 ~ $ cd ski.places
sgnorsoyan@dk8n80 ~/ski.places $ ls
equipment  newdir
sgnorsoyan@dk8n80 ~/ski.places $ mv newdir plans
sgnorsoyan@dk8n80 ~/ski.places $ ls
equipment  plans
sgnorsoyan@dk8n80 ~/ski.places $
```

Рис. 3.10: Создаю новый каталог

10. Определяю опции команды chmod и присваиваю права доступа файлам, которые описаны в лабе.

```
sgnorsoyan@dk8n80 ~/ski.places $ chmod --help
Использование: chmod [ПАРАМЕТР]... РЕЖИМ[,РЕЖИМ]... ФАЙЛ
или:      chmod [ПАРАМЕТР]... ВОСЬМЕРИЧНЫЙ-РЕЖИМ ФАЙЛ...
или:      chmod [ПАРАМЕТР]... --reference=ОФАЙЛ ФАЙЛ...
Смена РЕЖИМА доступа к указанным ФАЙЛАМ.
При задании --reference, установить режим
указанных ФАЙЛОВ как у ЭФАЙЛА.

-c, --changes          как verbose, но выводить только при выполнении
                        изменений
-f, --silent, --quiet  не выводить большинство сообщений об ошибках
-v, --verbose          выводить диагностические сообщения по каждому файлу
--no-preserve-root     не обрабатывать «/» особым образом (по умолчанию)
--preserve-root        отказываться рекурсивно обрабатывать «/»
--reference=ЭФАЙЛ      использовать режим доступа ЭФАЙЛА
                        вместо значений РЕЖИМ
-R, --recursive        рекурсивно изменять файлы и каталоги
--help                display this help and exit
--version              output version information and exit

Каждый РЕЖИМ задаётся в форме «[ugoа]*([-+]=([rwxXst]*[ugo]))+|([-+]=[0-7])+».

Страница справки по GNU coreutils: <https://www.gnu.org/software/coreutils/>
Об ошибках в переводе сообщений сообщайте по адресу <https://translationproject.org/team/ru.html>
Полная документация: <https://www.gnu.org/software/coreutils/chmod>
или доступная локально: info '(coreutils) chmod invocation'
sgnorsoyan@dk8n80 ~/ski.places $
```

Рис. 3.11: Определяю опции команды chmod

```
sgnorsoyan@dk8n80 ~ $ touch australia play my_os feathers
sgnorsoyan@dk8n80 ~ $ ls
abcl  australia  cb  chisla.cpp  feathers  home  may  monthly  my_os  parentdir  parentdir2  play  public.html  ski.places  work  Документы  Изображения  Общедоступные  c++
april  bin  cbr  equipment  GNUstep  lab7-1.asm  mkb  morefun  newdir  parentdir1  parentdir3  public  reports  tmp  Видео  Загрузки  Музыка  'Рабочий стол'  Шаблоны
sgnorsoyan@dk8n80 ~ $ ls -l australia play my_os feathers
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 australia
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 feathers
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 my_os
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 play
sgnorsoyan@dk8n80 ~ $ ls -l my_os
ls: невозможно получить доступ к 'my-os': Нет такого файла или каталога
sgnorsoyan@dk8n80 ~ $ ls -l my_os
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 my_os
sgnorsoyan@dk8n80 ~ $ chmod u+x my_os
sgnorsoyan@dk8n80 ~ $ ls -l my_os
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 my_os
sgnorsoyan@dk8n80 ~ $ ls -l feathers
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 feathers
sgnorsoyan@dk8n80 ~ $ chmod g+w feathers
sgnorsoyan@dk8n80 ~ $ ls -l feathers
-rw-rw-r-- 1 sgnorsoyan studsci 0 map 9 14:27 feathers
sgnorsoyan@dk8n80 ~ $ chmod u+x australia
sgnorsoyan@dk8n80 ~ $ ls -l australia
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 australia
chmod: неверный режим: «u+r»
По команде «chmod --help» можно получить дополнительную информацию.
sgnorsoyan@dk8n80 ~ $ ls -l australia
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 australia
sgnorsoyan@dk8n80 ~ $ ls -l play
-rw-r--r-- 1 sgnorsoyan studsci 0 map 9 14:27 play
sgnorsoyan@dk8n80 ~ $ chmod u+x play
sgnorsoyan@dk8n80 ~ $ ls -l play
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 play
sgnorsoyan@dk8n80 ~ $ chmod g-rw play
sgnorsoyan@dk8n80 ~ $ ls -l play
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 play
sgnorsoyan@dk8n80 ~ $ chmod o-r play
sgnorsoyan@dk8n80 ~ $ ls -l play
-rwxr--r-- 1 sgnorsoyan studsci 0 map 9 14:27 play
sgnorsoyan@dk8n80 ~ $
```

Рис. 3.12: желаю всем счастья

11. Выполню все действия в задании от 4.1. до 4.12.

```
sgnorsoyan@dk8n80 ~ $ cp feathers file.old
sgnorsoyan@dk8n80 ~ $ cp file.old play
sgnorsoyan@dk8n80 ~ $ cp play fun
sgnorsoyan@dk8n80 ~ $ cp fum play
cp: не удалось выполнить stat для 'fum': Нет такого файла или каталога
sgnorsoyan@dk8n80 ~ $ cp fun play
sgnorsoyan@dk8n80 ~ $ rm play games
rm: невозможно удалить 'games': Нет такого файла или каталога
sgnorsoyan@dk8n80 ~ $ mv play games
mv: не удалось выполнить stat для 'play': Нет такого файла или каталога
sgnorsoyan@dk8n80 ~ $
```

Рис. 3.13: выполняю задания

## 4 Контрольные вопросы

1. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th». 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. 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. • / – корень Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. Адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога. Только пользователь root имеет право читать и изменять файлы в этом каталоге. • /BIN – бинарные файлы пользователя Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления. • /SBIN – системные исполняемые



файлы Так же как и /bin, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог /usr. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя. • /ETC – конфигурационные файлы В этой папке содержатся конфигурационные файлы всех программ, установленных в системе. Кроме конфигурационных файлов, в системе инициализации Init Scripts, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ. • /DEV – файлы устройств В Linux все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны, камеры – это просто файлы в каталоге /dev/. Выполняется сканирование всех подключенных устройств и создание для них специальных файлов. • /PROC – информация о процессах По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его Pid, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. Также здесь можно найти информацию об использовании системных ресурсов. • /VAR – переменные файлы Название каталога /var говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кешы, базы данных и так далее. • /TMP – временные файлы В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию. • /USR – программы пользователя Это самый большой каталог с большим количеством функций. Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыку и документацию. • /HOME – домашняя папка В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т.д. • /BOOT – файлы загрузчика Содержит все файлы, связанные с загрузчиком системы. Это

ядро `vmlinuz`, образ `initrd`, а также файлы загрузчика, находящиеся в каталоге `/boot/grub`.

- `/LIB` – системные библиотеки Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах `/bin` и `/sbin`.
- `/OPT` – дополнительные программы В эту папку устанавливаются проприетарные программы, игры или драйвера. Это программы созданные в виде отдельных исполняемых файлов самими производителями.
- `/MNT` – монтирование В этот каталог системные администраторы могут монтировать внешние или дополнительные файловые системы.
- `/MEDIA` – съемные носители В этот каталог система монтирует все подключаемые внешние накопители –USB флешки, оптические диски и другие носители информации.
- `/SRV` – сервер В этом каталоге содержатся файлы серверов и сервисов.
- `/RUN` - процессы Каталог, содержащий PID файлы процессов, похожий на `/var/run`, но в отличие от него, он размещен в `TMPFS`, а поэтому после перезагрузки все файлы теряются.

3. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой `mount`.
4. 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. Файловую систему можно создать, используя команду `mkfs`. Ее краткое описание дано в пункте 5 в ходе выполнения заданий лабораторной работы.
6. Для просмотра текстовых файлов существуют следующие команды:
  - `cat` 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
  - `nl` Команда `nl` дей-

ствуется аналогично команде `cat`, но выводит еще и номера строк в столбце слева. • `less` 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 • `head` 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: `-c` (`-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 • `tail` 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: `-c` – 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. Утилита `sr` позволяет полностью копировать файлы и директории. Синтаксис: `sr` 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. 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 `-strip-trailing-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` – принудить

тельно перезаписывать существующие файлы

9. 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 # Выводы

В ходе выполнения лабораторной работы №5 я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. Так же я приобрела практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами, по проверке использования и обслуживания файловой системы.

## Список литературы

1. GNU Bash Manual [Электронный ресурс]. Free Software Foundation, 2016.  
URL: <https://www.gnu.org/software/bash/manual/>.
2. Newham C. Learning the bash Shell: Unix Shell Programming. O'Reilly Media, 2005. 354 с.
3. Zarrelli G. Mastering Bash. Packt Publishing, 2017. 502 с.
4. Robbins A. Bash Pocket Reference. O'Reilly Media, 2016. 156 с.
5. Таненбаум Э. Архитектура компьютера. 6-е изд. СПб.: Питер, 2013. 874 с.
6. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.