

Отчёт по лабораторной работе №5

дисциплина: Операционные системы

Максим Александрович Мишонков

Содержание

1	Цель работы	4
2	Теоретическое введение	5
3	Выполнение лабораторной работы	6
4	Выводы	19

Список иллюстраций

3.1	Копирование каталогов и файлов	6
3.2	Перемещение и переименование файлов и каталогов	7
3.3	Копирование файла в домашний каталог	7
3.4	Создание каталога и перемещение в него файла	7
3.5	Переименование файла	7
3.6	Создание файла в одном каталоге и копирование его в другой каталог с другим именем	8
3.7	Перемещение файлов	8
3.8	Перемещение каталога	8
3.9	Определение опций команды <code>chmod</code>	9
3.10	Просмотр содержимого файла	9
3.11	Создание каталогов, изменение прав владельца	10
3.12	Изменение прав владельца	10
3.13	Команда <code>mount</code>	11
3.14	Команда <code>fsck</code>	12
3.15	Команда <code>mkfs</code>	12
3.16	Команда <code>kill</code>	13
3.17	Команды	13

1 Цель работы

Целью данной лабораторной работы является ознакомление с файловой системой linux, 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 Теоретическое введение

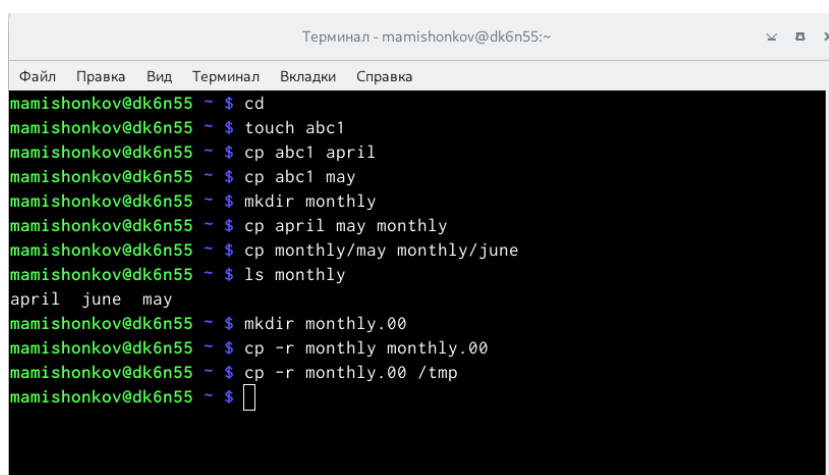
Файловая система (file system, ФС) — важная составляющая любой операционной системы (ОС), отвечающая за организацию, хранение, чтение, запись файлов. От ФС зависит физическая и логическая структура файлов, политика создания и управления ими, максимальный размер файла и длина его имени. Linux поддерживает множество разных file system, включая FAT, FAT32, NTFS из Windows. Но использовать рекомендуется «родные» системы: Ext3, Ext4, ReiserFS, XFS, Btrfs и пр.

Linux позволяет установить отдельную ФС для каждого раздела. Выбранная система определяет, как быстро будет выполняться работа с файлами, запись и чтение информации. Как в оперативной памяти будет храниться информация (и будет ли она вообще храниться), каким образом можно вносить изменения в конфигурацию ядра ОС — это тоже зависит от ФС.

File system Linux — пространство раздела, состоящее из кратных размеру сектора блоков. Обмен данными производится через VFS или с помощью драйверов. VFS (virtual file system) — это слой абстракции, необходимый для взаимодействия между ядром и софтом. VFS позволяет не думать о специфике работы той или иной ФС. Драйверы ФС обеспечивают взаимодействие между оборудованием (железом) и приложениями.

3 Выполнение лабораторной работы

1. Выполнил примеры, описанные в первой части лабораторной работы. Создал файл `abc1`, скопировал его в файл `april` и файл `may`. Создал каталог `monthly`, скопировал файлы `april` и `may` в каталог `monthly`. Скопировал файл `monthly/may` в файл с именем `june`. Создал каталог `monthly.00`, скопировал каталог `monthly` в каталог `monthly.00` и в катлог `/tmp`. (рис. [3.1])



```
Терминал - mamishonkov@dk6n55:~
Файл  Правка  Вид  Терминал  Вкладки  Справка
mamishonkov@dk6n55 ~ $ cd
mamishonkov@dk6n55 ~ $ touch abc1
mamishonkov@dk6n55 ~ $ cp abc1 april
mamishonkov@dk6n55 ~ $ cp abc1 may
mamishonkov@dk6n55 ~ $ mkdir monthly
mamishonkov@dk6n55 ~ $ cp april may monthly
mamishonkov@dk6n55 ~ $ cp monthly/may monthly/june
mamishonkov@dk6n55 ~ $ ls monthly
april  june  may
mamishonkov@dk6n55 ~ $ mkdir monthly.00
mamishonkov@dk6n55 ~ $ cp -r monthly monthly.00
mamishonkov@dk6n55 ~ $ cp -r monthly.00 /tmp
mamishonkov@dk6n55 ~ $
```

Рис. 3.1: Копирование каталогов и файлов

2. Изменил название файла `april` на `july` в домашнем каталоге. Переместил файл `july` в каталог `monthly.00`. Переименовал каталог `monthly.00` в `monthly.01`. Переместил каталог `monthly.00` в каталог `reports`. Переименовал каталог `reports/monthly.01` в `reports/monthly`. (рис. [3.2])

```

mamishonkov@dk6n55 ~ $ cd
mamishonkov@dk6n55 ~ $ mv april july
mamishonkov@dk6n55 ~ $ mv july monthly.00
mamishonkov@dk6n55 ~ $ ls monthly.00
july  monthly
mamishonkov@dk6n55 ~ $ mv monthly.00 monthly.01
mamishonkov@dk6n55 ~ $ mkdir reports
mamishonkov@dk6n55 ~ $ mv monthly.01 reports
mamishonkov@dk6n55 ~ $ mv reports/monthly.01 reports/monthly
mamishonkov@dk6n55 ~ $ clear

```

Рис. 3.2: Перемещение и переименование файлов и каталогов

3. Скопировал файл `/usr/include/sys/io.h` в домашний каталог и назвал его `equipment`. (рис. [3.3])

```

mamishonkov@dk6n55 ~ $ cp /usr/include/sys/io.h /afs/.dk.sci.pfu.edu.ru/home/m/a/mamishonkov/equipment
mamishonkov@dk6n55 ~ $ ls
abcl    may    public  work    Загрузки  Общедоступные  Шаблоны
bin     monthly public_html Видео    Изображения  'Отчёт 4 Markdown и скриншоты'
equipment newdir  reports  Документы Музыка      'Рабочий стол'

```

Рис. 3.3: Копирование файла в домашний каталог

4. В домашнем каталоге создал директорию `ski.places`. Переместил файл `equipment` в созданный каталог. (рис. [3.4])

```

mamishonkov@dk6n55 ~ $ mkdir ski.places
mamishonkov@dk6n55 ~ $ cp equipment ski.places
mamishonkov@dk6n55 ~ $ mv equipment ~/ski.places
mamishonkov@dk6n55 ~ $ ls ski.places
equipment
mamishonkov@dk6n55 ~ $

```

Рис. 3.4: Создание каталога и перемещение в него файла

5. Переименовал файл `~/ski.places/equipment` в `~/ski.places/equiplist`. (рис. [3.5])

```

mamishonkov@dk6n55 ~ $ mv ~/ski.places/equipment ~/ski.places/equiplist
mamishonkov@dk6n55 ~ $ ls ski.places
equiplist
mamishonkov@dk6n55 ~ $

```

Рис. 3.5: Переименование файла

6. Создал в домашнем каталоге файл `abc1` и скопировал его в каталог `~/ski.plases`, назвав его `equiplist2`. (рис. [3.6])

```
mamishonkov@dk6n55 ~$ touch abc1
mamishonkov@dk6n55 ~$ cp abc1 ~/ski.plases
mamishonkov@dk6n55 ~$ mv ~/ski.plases/abc1 ~/ski.plases/equiplist2
mamishonkov@dk6n55 ~$ ls ski.plases
equiplist  equiplist2
mamishonkov@dk6n55 ~$
```

Рис. 3.6: Создание файла в одном каталоге и копирование его в другой каталог с другим именем

7. Создал каталог `equiplist` в каталоге `~/ski.plases`. Переместил файлы `equiplist` и `equiplist2` в каталог `equipment`. (рис. [3.7])

```
mamishonkov@dk6n55 ~$ mkdir ~/ski.plases/equipment
mamishonkov@dk6n55 ~$ ls ski.plases
equiplist  equiplist2  equipment
mamishonkov@dk6n55 ~$ mv ~/ski.plases/equiplist ~/ski.plases/equipment
mamishonkov@dk6n55 ~$ mv ~/ski.plases/equiplist2 ~/ski.plases/equipment
mamishonkov@dk6n55 ~$ ls ~/ski.plases/equipment
equiplist  equiplist2
mamishonkov@dk6n55 ~$
```

Рис. 3.7: Перемещение файлов

8. Создал каталог `newdir`. Переместил его в каталог `ski.plases` и назвал его `plans`. (рис. [3.8])

```
mamishonkov@dk6n55 ~$ mkdir newdir
mamishonkov@dk6n55 ~$ mv newdir plans
mamishonkov@dk6n55 ~$ mv plans ski.plases
mamishonkov@dk6n55 ~$ ls ski.plases
equipment  plans
mamishonkov@dk6n55 ~$
```

Рис. 3.8: Перемещение каталога

9. Определил опции команды `chmod`, необходимые для того, чтобы присвоить файлам права доступа, считая, что в начале таких прав нет. (рис. [3.9])


```

mamishonkov@dk6n55 ~ $ mkdir australia
mamishonkov@dk6n55 ~ $ mkdir play
mamishonkov@dk6n55 ~ $ touch my_os
mamishonkov@dk6n55 ~ $ touch feathers
mamishonkov@dk6n55 ~ $ chmod 744 australia
mamishonkov@dk6n55 ~ $ chmod 711 play
mamishonkov@dk6n55 ~ $ chmod 544 my_os
mamishonkov@dk6n55 ~ $ chmod 664 feathers
mamishonkov@dk6n55 ~ $ ls -l
итого 35
-rw-r--r-- 1 mamishonkov studsci 0 map 6 18:48 abc1
drwxr--r-- 2 mamishonkov studsci 2048 map 6 19:07 australia
drwxr-xr-x 2 mamishonkov studsci 2048 фев 21 12:57 bin
-rw-rw-r-- 1 mamishonkov studsci 0 map 6 19:07 feathers
-rw-r--r-- 1 mamishonkov studsci 0 map 6 18:24 may
drwxr-xr-x 2 mamishonkov studsci 2048 map 6 18:27 monthly
-r-xr--r-- 1 mamishonkov studsci 0 map 6 19:07 my_os
drwx--x--x 2 mamishonkov studsci 2048 map 6 19:07 play
drwxr-xr-x 3 mamishonkov root 2048 фев 4 21:56 public
lrwxr-xr-x 1 mamishonkov root 18 map 2 23:37 public_html -> public/public_html
drwxr-xr-x 3 mamishonkov studsci 2048 map 6 18:32 reports
drwxr-xr-x 4 mamishonkov studsci 2048 map 6 19:00 ski.places
drwxr-xr-x 5 mamishonkov studsci 2048 фев 21 13:43 work
drwxr-xr-x 2 mamishonkov studsci 2048 сен 8 11:02 Видео
drwxr-xr-x 2 mamishonkov studsci 2048 сен 8 11:02 Документы
drwxr-xr-x 4 mamishonkov studsci 2048 мар 2 14:19 Загрузки
drwxr-xr-x 3 mamishonkov studsci 2048 сен 15 11:34 Изображения
drwxr-xr-x 2 mamishonkov studsci 2048 сен 8 11:02 Музыка
drwxr-xr-x 2 mamishonkov studsci 2048 сен 8 11:02 Общедоступные
drwxr-xr-x 2 mamishonkov studsci 2048 окт 28 20:25 'Отчёт 4 Markdown и скриншоты'
drwxr-xr-x 2 mamishonkov studsci 2048 сен 15 11:55 'Рабочий стол'
drwxr-xr-x 2 mamishonkov studsci 2048 сен 8 11:02 Шаблоны

```

Рис. 3.9: Определение опций команды chmod

10. Просмотрел содержимое файла /etc/passwd. (рис. [3.10])

```

mamishonkov@dk6n55 ~ $ cat /etc/passwd
root:x:0:0:System user; root:/root:/bin/bash
bin:x:1:1:bin:/bin:/bin/false
daemon:x:2:2:daemon:/sbin:/bin/false
adm:x:3:4:adm:/var/adm:/bin/false
lp:x:4:7:lp:/var/spool/lpd:/bin/false
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 program user:/var/spool/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/bin/false
uucp:x:10:14:uucp:/var/spool/uucppublic:/bin/false
operator:x:11:0:operator:/root:/bin/bash
man:x:13:15:System user; man:/dev/null:/sbin/nologin
postmaster:x:14:12:Postmaster user:/var/spool/mail:/sbin/nologin
cron:x:16:16:A user for sys-process/cronbase:/var/spool/cron:/sbin/nologin
ftp:x:21:21:./home/ftp:/bin/false
sshd:x:22:22:User for ssh:/var/empty:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/bin/false
squid:x:31:31:Squid:/var/cache/squid:/bin/false
gdm:x:32:32:User for running GDM:/var/lib/gdm:/sbin/nologin
xfs:x:33:33:X Font Server:/etc/X11/fs:/bin/false

```

Рис. 3.10: Просмотр содержимого файла

11. Скопировал файл ~/feathers в файл ~/file.old. Переместил файл ~/file.old В

катлог ~/play. Скопировал каталог ~/play в каталог ~/fun. Переместил каталог ~/fun в каталог ~/play и назвал его games. Лишил владельца файла feathers прав на чтение. Если мы попытаемся просмотреть файл feathers командой cat, то получим отказ в доступе, так как мы лишили владельца прав на чтение. Если мы попытаемся скопировать файл feathers, то также получим отказ. (рис. [3.11])

```
mamishonkov@dk6n55 ~$ cp feathers file.old
mamishonkov@dk6n55 ~$ mv file.old play
mamishonkov@dk6n55 ~$ cp -r play fun
mamishonkov@dk6n55 ~$ mv fun play
mamishonkov@dk6n55 ~$ mv play/fun games
mamishonkov@dk6n55 ~$ ls fun
ls: невозможно получить доступ к 'fun': Нет такого файла или каталога
mamishonkov@dk6n55 ~$ grep fun
^Z
[1]+  Остановлен  grep --colour=auto fun
mamishonkov@dk6n55 ~$ cp -r play fun
mamishonkov@dk6n55 ~$ ls
abc1      fun      my_os      reports    Документы  Общедоступные
australia games    play       ski.places Загрузки   'Отчёт 4 Markdown и скриншоты'
bin       may      public      work       Изображения 'Рабочий стол'
feathers  monthly public_html Видео       Музыка      Шаблоны
mamishonkov@dk6n55 ~$ mv fun play
mamishonkov@dk6n55 ~$ mv play/fun play/games
mamishonkov@dk6n55 ~$ chmod u-r feathers
mamishonkov@dk6n55 ~$ cat feathers
cat: feathers: Отказано в доступе
mamishonkov@dk6n55 ~$ cp feathers monthly
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
```

Рис. 3.11: Создание каталогов, изменение прав владельца

12. Дал владельцу файла feathers право на чтение, лишил владельца каталога play права на выполнение. При переходе в него получил отказ в доступе. Дал владельцу каталога play право на выполнение. (рис. [3.12])

```
mamishonkov@dk6n55 ~$ chmod u+r feathers
mamishonkov@dk6n55 ~$ chmod u-r play
mamishonkov@dk6n55 ~$ cd play
mamishonkov@dk6n55 ~/play$ cd ~
mamishonkov@dk6n55 ~$ chmod u+x play
mamishonkov@dk6n55 ~$
```

Рис. 3.12: Изменение прав владельца

13. Прочитал man по командам mount, fsck, mkfs, kill. Команда mount предназначена для монтирования файловой системы. Она служит для подключения файловых систем разных устройств к дереву. Например, команда

“mount -t vfstype device dir” предлагает ядру смонтировать файловую систему указанного типа, расположенную на определённом устройстве, к заданному каталогу. Команда fsck - утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Например, если нужно восстановить файловую систему на неотором устройстве /dev/sdb2, следует воспользоваться командой “sudo fsck -y/dev/sdb2”. Команда mkfs создаёт новую файловую систему Linux. Например, команда “mkfs -t ext/2/dev/hdb1” создаёт файловую систему типа ext 2 в разделе /dev/hdb1. Команда kill посылает сигнал процессу или выводит список допустимых сигналов. Например. команда “lill -KILL 3121” посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс. (рис. [3.13], [3.14], [3.15], [3.16], [3.17])

```
MOUNT(8)                                     System Administration                                     MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

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

    mount [-fnrsvw] [-o options] device|mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount --make-[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable] mountpoint

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

Manual page mount(8) line 1 (press h for help or q to quit)
```

Рис. 3.13: Команда mount

```

FSCK(8)                                System Administration                                FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

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

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

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

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

    0
    No errors

Manual page fsck(8) line 1 (press h for help or q to quit)

```

Рис. 3.14: Команда fsck

```

MKFS(8)                                System Administration                                MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard disk partition. The
    device argument is either the device name (e.g., /dev/hda1, /dev/sdb2), or a regular file that
    shall contain the filesystem. The size argument is the number of blocks to be used for the
    filesystem.

    The exit status returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem builders (mkfs.fstype)
    available under Linux. The filesystem-specific builder is searched for via your PATH
    environment setting only. Please see the filesystem-specific builder manual pages for further
    details.

Manual page mkfs(8) line 1 (press h for help or q to quit)

```

Рис. 3.15: Команда mkfs

```
KILL(1)                                User Commands                                KILL(1)

NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available signals. Particularly
    useful signals include HUP, INT, KILL, STOP, CONT, and 0. Alternate signals may be specified
    in three ways: -9, -SIGKILL or -KILL. Negative PID values may be used to choose whole process
    groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all
    processes except the kill process itself and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
    --signal <signal>
        Specify the signal to be sent. The signal can be specified by using name or number.

Manual page kill(1) line 1 (press h for help or q to quit)
```

Рис. 3.16: Команда kill

```
mamishonkov@dk6n55 ~ $ man mount
[2]+  Остановлен   man mount
mamishonkov@dk6n55 ~ $ man fsck
[3]+  Остановлен   man fsck
mamishonkov@dk6n55 ~ $ man mkfs
[4]+  Остановлен   man mkfs
mamishonkov@dk6n55 ~ $ man kill
[5]+  Остановлен   man kill
mamishonkov@dk6n55 ~ $
```

Рис. 3.17: Команды

Ответы на контрольные вопросы:

1).Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df-Th». Из рисунка видно, что на моем компьютере есть следующие файловые системы: dev tmpfs,tmpfs,ext4,iso9660. dev tmpfs 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.

"/BIN" – бинарные файлы пользователя. Этот каталог содержит исполняемые файлы. Здесь находятся все бинарные файлы, которые используются для запуска программ.

"/SBIN" – системные исполняемые файлы. Так же как и "/bin", содержит двоичные исполняемые файлы, но для системных нужд.

"/ETC" – конфигурационные файлы. В этой папке содержатся конфигурационные файлы всех системных программ.

"/DEV" – файлы устройств. В Linux все, в том числе внешние устройства являются файлами.

"/PROC" – информация о процессах. По сути, это псевдофайловая система, содержащая информацию о процессах.

"/VAR" – переменные файлы. Название каталога "/var" говорит само за себя, он должен содержать переменные файлы.

"/TMP" – временные файлы. В этом каталоге содержатся временные файлы, созданные системой.

"/USR" – программы пользователя. Это самый большой каталог с большим количеством файлов.

"/HOME" – домашняя папка. В этой папке хранятся домашние каталоги всех пользователей.

"/BOOT" – файлы загрузчика. Содержит все файлы, связанные с загрузчиком системы. Здесь находятся файлы, которые используются для загрузки системы.

"/LIB" – системные библиотеки. Содержит файлы системных библиотек, которые используются для запуска программ.

"/OPT" – дополнительные программы. В эту папку устанавливаются проприетарные программы.

"/MNT" – монтирование. В этот каталог системные администраторы могут монтировать файловые системы.

"/MEDIA" – съемные носители. В этот каталог система монтирует все подключаемые внешние носители.

"/SRV" – сервер. В этом каталоге содержатся файлы серверов и сервисов.

"/RUN" – процессы. Каталог, содержащий PID файлы процессов, похожий на "/var/run", но для процессов.

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 последнем пункте 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.
`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` действует аналогично команде `cat`, но выводит еще номера строк в столбец.
`less`. Существенно более развитая команда для пролистывания текста. При чтении данных с терминала:
`g` –при поиске подсвечивать только текущее найденное слово (по умолчанию подсвечивать все найденные слова)
`N` –показывать номера строк
`head`. Команда `head` выводит начальные строки (по умолчанию – 10) из одного или нескольких файлов.
`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 строки
`tail` Эта команда позволяет выводить заданное количество строк с конца файла, а также с конца нескольких файлов.
`c` –выводить указанное количество байт с конца файла–`f` –обновлять информацию по мере поступления данных
`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 `cp` позволяет полностью копировать файлы и директории. Синтаксис: `cp [опции] файл-источник файл-приемник`. После выполнения команды файл-источник будет полностью перенесен в файл-приемник. Если в конце указан слэш, файл будет записан в заданную директорию с оригинальным именем. Основные опции:–`attributes-only` –не копировать содержимое файла, а только флаги доступа и владельца–`f`, `--force` –перезаписывать существующие файлы–`i`, `--interactive` –спрашивать, нужно ли перезаписывать существующие файлы–`L` –копировать

не символические ссылки, а то, на что они указывают -n –не перезаписывать существующие файлы-P –не следовать символическим ссылкам-r –копировать папку Linux рекурсивно-s –не выполнять копирование файлов в Linux, а создавать символические ссылки-u –скопировать файл, только если он был изменён-x –не выходить за пределы этой файловой системы-r –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` –принудительно 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 мо-

жет 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 `чтени` `w` запись `x` выполнение `u` (user) владелец файла `g` (group) группа, к которой принадлежит владелец файла `o` (others) все остальные.

4 Выводы

В ходе выполнения данной лабораторной работы я ознакомился с файловой системой linux, именами и содержанием каталогов, приобрёл практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами, по проверке использования диска и обслуживанию файловой системы.