

Отчет по лабораторной работе №5

дисциплина: Операционные системы

Егорова Юлия Владимировна

Содержание

1	Цель работы	4
2	Выполнение лабораторной работы	5
3	Контрольные вопросы	15
4	Вывод	22

Список иллюстраций

2.1	Копирование файла в файлы.	5
2.2	Копирование файлов в каталог.	5
2.3	Копирование файлов в произвольный каталог.	5
2.4	Создание каталога <code>monthly.00</code> и копирование каталога <code>monthly</code> . .	6
2.5	Копирование каталога <code>monthly.00</code>	6
2.6	Изменение названия файла.	6
2.7	Перемещение файла <code>july</code> в каталог <code>monthly.00</code>	6
2.8	Переименовывание каталога.	6
2.9	Перемещение одного каталога в другой.	6
2.10	Изменение названия каталога.	7
2.11	Создание файла.	7
2.12	Лишение права на выполнение.	7
2.13	Создание каталога.	7
2.14	Запрет на чтение.	7
2.15	Создание файла.	7
2.16	Копирование файла в домашний каталог и изменение названия. .	8
2.17	Создание директории.	8
2.18	Перемещение файла в каталог.	8
2.19	Переименовывание файла.	8
2.20	Создание файла.	8
2.21	Копирование файла и изменение его названия.	8
2.22	Создание каталога.	9
2.23	Перемещение файлов в каталог.	9
2.24	Создание, перемещение и изменение названия каталога.	9
2.25	Определение опций команды <code>chmod</code>	9
2.26	Просмотр содержимого с помощью команды <code>cat</code>	10
2.27	Копирование одного файла в другой.	10
2.28	Перемещение файла в каталог.	10
2.29	Копирование одного каталога в другой	10
2.30	Перемещение каталога, изменение названия и прав доступа. . . .	11
2.31	Описание команды <code>mount</code>	12
2.32	Описание команды <code>fsck</code>	12
2.33	Описание команды <code>kill</code>	13
2.34	Описание команды <code>mkfs</code>	14

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 Выполнение лабораторной работы

1) Выполняем все примеры, приведенные в первой части описания лабораторной работы:

1.1) Копируем в файл в текущий каталог. Копируем файл abc1 в файл april и файл may:

```
yvegorova@dk6n62 ~ $ cd
yvegorova@dk6n62 ~ $ touch abc1
yvegorova@dk6n62 ~ $ cp abc1 april
yvegorova@dk6n62 ~ $ cp abc1 may
```

Рис. 2.1: Копирование файла в файлы.

1.2) Копируем несколько файлов в каталог. Копируем файлы april и may в каталог monthly:

```
yvegorova@dk6n62 ~ $ mkdir monthly
yvegorova@dk6n62 ~ $ cp april may monthly
```

Рис. 2.2: Копирование файлов в каталог.

1.3) Копируем файлы в произвольном каталоге. Копируем файл monthly/may в файл с именем june:

```
yvegorova@dk6n62 ~ $ cp monthly/may monthly/june
yvegorova@dk6n62 ~ $ ls monthly
april june may
```

Рис. 2.3: Копирование файлов в произвольный каталог.

1.4) Копируем каталог monthly в каталог monthly.00

```
yvegorova@dk6n62 ~ $ mkdir monthly.00
yvegorova@dk6n62 ~ $ cp -r monthly monthly.00
```

Рис. 2.4: Создание каталога monthly.00 и копирование каталога monthly.

1.5) Копируем каталог monthly.00 в каталог /tmp

```
yvegorova@dk6n62 ~ $ cp -r monthly.00 /tmp
yvegorova@dk6n62 ~ $ cd
```

Рис. 2.5: Копирование каталога monthly.00.

1.6) Изменяем название файла april на july в домашнем каталоге:

```
yvegorova@dk6n62 ~ $ cd
yvegorova@dk6n62 ~ $ mv april july
```

Рис. 2.6: Изменение названия файла.

1.7) Перемещаем файл july в каталог monthly.00:

```
yvegorova@dk6n62 ~ $ mv july monthly.00
yvegorova@dk6n62 ~ $ ls monthly.00
july  monthly
```

Рис. 2.7: Перемещение файла july в каталог monthly.00.

1.8) Переименовываем каталог monthly.00 в каталог в monthly.01:

```
yvegorova@dk6n62 ~ $ mv monthly.00 monthly.01
```

Рис. 2.8: Переименовывание каталога.

1.9) Перемещаем каталог monthly.01 в каталог reports:

```
yvegorova@dk6n62 ~ $ mkdir reports
yvegorova@dk6n62 ~ $ mv monthly.01 reports
```

Рис. 2.9: Перемещение одного каталога в другой.

1.10) Переименовываем каталог reports/monthly.01 в reports/monthly:

```
yvegorova@dk6n62 ~ $ mv reports/monthly.01 reports
yvegorova@dk6n62 ~ $ mv reports/monthly.01 reports/monthly
```

Рис. 2.10: Изменение названия каталога.

1.11) Создаем файл ~/may с правом выполнения для владельца. Лишаем владельца файла ~/may права на выполнение:

```
yvegorova@dk6n62 ~ $ cd
yvegorova@dk6n62 ~ $ touch may
yvegorova@dk6n62 ~ $ ls -l may
-rw-r--r-- 1 yvegorova studsci 0 мая  4 14:47 may
yvegorova@dk6n62 ~ $ chmod u+x may
yvegorova@dk6n62 ~ $ ls -l may
-rwxr--r-- 1 yvegorova studsci 0 мая  4 14:47 may
```

Рис. 2.11: Создание файла.

```
yvegorova@dk6n62 ~ $ chmod u-x may
yvegorova@dk6n62 ~ $ ls -l may
-rw-r--r-- 1 yvegorova studsci 0 мая  4 14:47 may
```

Рис. 2.12: Лишение права на выполнение.

1.12) Создаем каталог monthly с запретом на чтение для членов группы и всех остальных пользователей:

```
yvegorova@dk6n62 ~ $ cd
yvegorova@dk6n62 ~ $ mkdir monthly
```

Рис. 2.13: Создание каталога.

```
yvegorova@dk6n62 ~ $ chmod g-r,o-r monthly
```

Рис. 2.14: Запрет на чтение.

1.13) Создаем файл ~/abc1 с правом записи для членов группы:

```
yvegorova@dk6n62 ~ $ cd
yvegorova@dk6n62 ~ $ touch abc1
yvegorova@dk6n62 ~ $ chmod g+w abc1
```

Рис. 2.15: Создание файла.

2) Копируем файл /usr/include/sys/io.h в домашний каталог и называем его equipment:

```
yvegorova@dk6n62 ~ $ cp /usr/include/sys/io.h equipment
```

Рис. 2.16: Копирование файла в домашний каталог и изменение названия.

2.1) Создаем директорию ~/ski.places. Перемещаем файл equipment в каталог ~/ski.places:

```
yvegorova@dk6n62 ~ $ mkdir ski.places
```

Рис. 2.17: Создание директории.

```
yvegorova@dk6n62 ~ $ mv equipment ski.places
```

Рис. 2.18: Перемещение файла в каталог.

2.2) Переименовываем файл ~/ski.places/equipment в ~/ski.places/equiplist:

```
yvegorova@dk6n62 ~ $ cd  
yvegorova@dk6n62 ~ $ mv ski.places/equipment ski.places/equiplist
```

Рис. 2.19: Переименовывание файла.

2.3) Создаем в домашнем каталоге файл abc1 и копируем его в каталог ~/ski.places, называем его equiplist2:

```
yvegorova@dk6n62 ~ $ touch abc1
```

Рис. 2.20: Создание файла.

```
yvegorova@dk6n62 ~ $ cp abc1 ski.places  
yvegorova@dk6n62 ~ $ mv ski.places/abc1 ski.places/equiplist2
```

Рис. 2.21: Копирование файла и изменение его названия.

2.4) Создаем каталог с именем equipment в каталоге ~/ski.places:


```
yvegorova@dk6n62 ~ $ mkdir ski.places/equipment
```

Рис. 2.22: Создание каталога.

2.5) Перемещаем файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment:

```
yvegorova@dk6n62 ~ $ mv ski.places/equiplist ski.places/equipment
yvegorova@dk6n62 ~ $ mv ski.places/equiplist2 ski.places/equipment
```

Рис. 2.23: Перемещение файлов в каталог.

2.6) Создаем и перемещаем каталог ~/newdir в каталог ~/ski.places и называем его plans:

```
yvegorova@dk6n62 ~ $ mkdir newdir
yvegorova@dk6n62 ~ $ mv newdir plans
yvegorova@dk6n62 ~ $ mv plans ski.places
yvegorova@dk6n62 ~ $ ls
. bin 'id -un' monthly public reports tmp yvegorova Документы Изображения Общедоступные Рабочий стол
abc1 course-directory-student-template may os-intro public_html ski.places work Видео Загрузки Музыка
```

Рис. 2.24: Создание, перемещение и изменение названия каталога.

3) Определяем опции команды chmod, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет. Предварительно создаем файлы:

```
yvegorova@dk6n62 ~ $ mkdir australia
yvegorova@dk6n62 ~ $ mkdir play
yvegorova@dk6n62 ~ $ touch my_os
yvegorova@dk6n62 ~ $ touch feathers
yvegorova@dk6n62 ~ $ chmod 744 australia
yvegorova@dk6n62 ~ $ chmod 711 play
yvegorova@dk6n62 ~ $ chmod 544 my_os
yvegorova@dk6n62 ~ $ chmod 664 feathers
yvegorova@dk6n62 ~ $ ls -l
итого 66
-rw-r--r-- 1 yvegorova studsci 5086 мая 4 15:16 -
-rw-rw-r-- 1 yvegorova studsci 0 мая 4 15:32 abc1
drwxr--r-- 2 yvegorova studsci 2048 мая 4 16:07 australia
drwxr-xr-x 2 yvegorova studsci 2048 апр 30 13:56 bin
drwxr-xr-x 5 yvegorova studsci 2048 апр 22 18:14 course-directory-student-template
-rw-rw-r-- 1 yvegorova studsci 0 мая 4 16:08 feathers
drwxr-xr-x 2 yvegorova studsci 2048 апр 21 13:22 'id -un'
-rw-r--r-- 1 yvegorova studsci 0 мая 4 14:47 may
drwx--x--x 2 yvegorova studsci 2048 мая 4 14:40 monthly
-r-xr--r-- 1 yvegorova studsci 0 мая 4 16:07 my_os
drwxr-xr-x 3 yvegorova studsci 2048 апр 22 18:11 os-intro
drwx--x--x 2 yvegorova studsci 2048 мая 4 16:07 play
drwxr-xr-x 3 yvegorova studsci 2048 сен 1 2021 public
lrwxr-xr-x 1 yvegorova root 18 апр 14 23:57 public_html -> public/public_html
drwxr-xr-x 3 yvegorova studsci 2048 мая 4 14:45 reports
drwxr-xr-x 4 yvegorova studsci 2048 мая 4 15:58 ski.places
drwxr-xr-x 2 yvegorova studsci 2048 апр 29 15:07 tmp
drwxr-xr-x 4 yvegorova studsci 2048 апр 30 13:38 work
drwxr-xr-x 2 yvegorova studsci 2048 сен 2 2021 yvegorova
drwxr-xr-x 2 yvegorova studsci 2048 окт 14 2021 Видео
drwxr-xr-x 3 yvegorova studsci 4096 апр 30 13:58 Загрузки
drwxr-xr-x 2 yvegorova studsci 16384 апр 30 15:02 Изображения
drwxr-xr-x 2 yvegorova studsci 2048 сен 2 2021 Музыка
drwxr-xr-x 2 yvegorova studsci 2048 сен 2 2021 Общедоступные
drwxr-xr-x 2 yvegorova studsci 4096 окт 27 2021 'Рабочий стол'
drwxr-xr-x 2 yvegorova studsci 2048 сен 2 2021 Шаблоны
```

Рис. 2.25: Определение опций команды chmod.

4) Просматриваем содержимое файла /etc/passwd:

```
yvegorova@dk6n62 ~ $ 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
games:x:35:35:games:/usr/games:/bin/bash
named:x:40:40:bind:/var/bind:/bin/false
mysql:x:60:60:MySQL program user:/dev/null:/sbin/nologin
postgres:x:70:70:PostgreSQL program user:/var/lib/postgresql:/bin/sh
nut:x:84:84:nut:/var/state/nut:/bin/false
cyrus:x:85:12::/usr/cyrus:/bin/false
vpopmail:x:89:89::/var/vpopmail:/bin/false
alias:x:200:200::/var/qmail/alias:/bin/false
qmaild:x:201:200::/var/qmail:/bin/false
qmail1:x:202:200::/var/qmail:/bin/false
qmailp:x:203:200::/var/qmail:/bin/false
qmailq:x:204:201::/var/qmail:/bin/false
qmailr:x:205:201::/var/qmail:/bin/false
qmails:x:206:201::/var/qmail:/bin/false
postfix:x:207:207:postfix:/var/spool/postfix:/bin/false
smmsp:x:209:209:smmsp:/var/spool/queue:/bin/false
portage:x:250:250:System user; portage:/var/tmp/portage:/sbin/nologin
nobody:x:65534:65534:System user; nobody:/dev/null:/sbin/nologin
sockd:x:101:214:A user for net-proxy/dante:/dev/null:/sbin/nologin
rpc:x:111:111:added by portage for portmap:/dev/null:/bin/false
ntp:x:123:123:user for ntp daemon:/dev/null:/sbin/nologin
ldap:x:439:439:added by portage for openldap:/usr/lib/openldap:/bin/false
messagebus:x:102:440:System user; messagebus:/dev/null:/sbin/nologin
haldaemon:x:103:441:added by portage for hal:/dev/null:/bin/false
dhcp:x:104:10148:user for dhcp daemon:/dev/null:/sbin/nologin
ivman:x:105:46:added by portage for ivman:/dev/null:/bin/false
hsqldb:x:106:10161:User for hsqldb:/dev/null:/sbin/nologin
beagleindex:x:107:10225:added by portage for beagle:/var/lib/cache/beagle:/usr/sbin/nologin
avahi:x:108:10227:user for avahi:/dev/null:/sbin/nologin
avahi-autoipd:x:109:10228:user for avahi-autoipd:/dev/null:/sbin/nologin
```

Рис. 2.26: Просмотр содержимого с помощью команды cat.

4.1) Копируем файл ~/feathers в файл ~/file.old:

```
yvegorova@dk6n62 ~ $ cp feathers file.old
```

Рис. 2.27: Копирование одного файла в другой.

4.2) Перемещаем файл ~/file.old в каталог ~/play:

```
yvegorova@dk6n62 ~ $ mv file.old play
```

Рис. 2.28: Перемещение файла в каталог.

4.3) Копируем каталог ~/play в каталог ~/fun:

```
yvegorova@dk6n62 ~ $ cp -r play fun
```

Рис. 2.29: Копирование одного каталога в другой

4.4) Перемещаем каталог ~/fun в каталог ~/play и называем его games. Затем лишаем владельца файла ~/feathers права на чтение, просматриваем файл командой cat и даем владельцу право на чтение. Потом лишаем владельца каталога ~/play права на выполнение, заходим в каталог и возвращаем право на выполнение.

```
yvegorova@dk6n62 ~ $ mv fun play
yvegorova@dk6n62 ~ $ mv play/fun play/games
yvegorova@dk6n62 ~ $ chmod u-r feathers
yvegorova@dk6n62 ~ $ chmod u+r feathers
yvegorova@dk6n62 ~ $ chmod u-r play
yvegorova@dk6n62 ~ $ cd play
yvegorova@dk6n62 ~/play $ cd
yvegorova@dk6n62 ~ $ chmod u-x play
yvegorova@dk6n62 ~ $ cd play
yvegorova@dk6n62 ~/play $ cd
yvegorova@dk6n62 ~ $ chmod u+x play
yvegorova@dk6n62 ~ $ chmod u-x play
yvegorova@dk6n62 ~ $ chmod u+r play
```

Рис. 2.30: Перемещение каталога, изменение названия и прав доступа.

5) По команде man читаем описание команд mount, mkfs, fsck, kill:

Команда mount:

Предназначена для монтирования файловой системы. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого дерева обозначается как /. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву. Наиболее часто встречающаяся форма команды mount выглядит следующим образом: «mount -t vfstype device dir» Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типа vfstype, расположенную на устройстве device, к заданному каталогу dir, который часто называют точкой монтирования

Посылает сигнал процессу или выводит список допустимых сигналов. Имеет следующий синтаксис: `kill [опции] PID`, где PID – это PID (числовой идентификатор) процесса или несколько PID процессов, если требуется послать сигнал сразу нескольким процессам. Например, команда «`kill -KILL 3121`» посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс

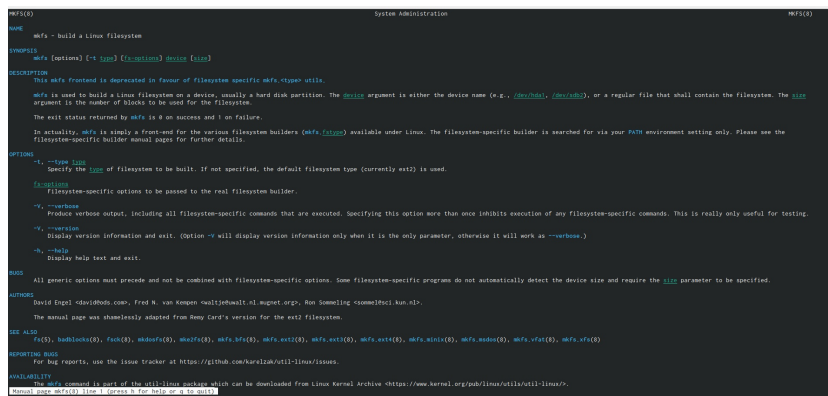


Рис. 2.33: Описание команды `kill`

Команда `mkfs`:

Создаёт новую файловую систему Linux. Имеет следующий синтаксис: `mkfs [-V] [-t fstype] [fs-options] filesys [blocks]` `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 (например, `/dev/hda1`, `/dev/sdb2`) или точка монтирования (например, `/`, `/usr`, `/home`). Аргументом `blocks` указывается количество блоков, которые выделяются для использования этой файловой системой. По окончании работы `mkfs` возвращает 0 - в случае успеха, а 1 - при неудачной операции. Например, команда «`mkfs -t ext2 /dev/hdb1`» создаёт файловую систему типа `ext2` в разделе `/dev/hdb1` (второй жёсткий диск)

```

kill(1)                                User Commands                                kill(1)
NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] qpid [...]

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 SIGKILL. Alternate signals may be specified in three ways: -s, -SIGKILL or -KILL.
    Negative PID values may be used to choose whole process groups; see the PIDs column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.

OPTIONS
    -qpid [...]
        Send signal to every qpid listed.

    -s signal
    -s signal
        Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

    -s name value
        Use signame() rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t structure.

    -l, --list [signal]
        List signal names. This option has optional argument, which will convert signal number to signal name, or other way round.

    -L, --table
        List signal names in a nice table.

NOTES
    Your shell (command line interpreter) may have a built-in kill command. You may need to run the command described here as /bin/kill to solve the conflict.

EXAMPLES
    kill -9 -1
        Kill all processes you can kill.

    kill -1 11
        Translate number 11 into a signal name.

    kill -L
        List the available signal choices in a nice table.

    kill 123 456 2345 3456
        Send the default signal, SIGTERM, to all these processes.

SEE ALSO
    kill(2), killall(1), nice(1), ps(1), renice(1), signal(7), signame(2), shkill(1)

TRANSLATION
    This command meets appropriate standards. The -s flag is Linux-specific.
    Manual page kill(1) from Linux 4.10.0-rc4-1404.el7.x86_64 #10-2017

```

Рис. 2.34: Описание команды mkfs

3 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: 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 ТБ;
- максимальный размер раздела: 16 ТБ;
- 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 CDROM. 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.(рис.2.22)

/ – 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` действует 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` – ис-

пользуется 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 -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 — 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 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

4 Вывод

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