

Отчет по лабораторной работе №7

Дисциплина: Операционные системы

Кириянова Екатерина Андреевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
4.1	Выполнение примеров из первой части описания лабораторной работы	9
4.2	Работа с файлами	11
4.3	Изменение прав доступа	13
4.4	Работа с файлами и правами доступа	13
4.5	Изучение команд с помощью man	15
5	Выводы	18
	Список литературы	19

Список иллюстраций

4.1	Пример 1	9
4.2	Пример 2	9
4.3	Пример 3	9
4.4	Пример 4	9
4.5	Пример 5	10
4.6	Пример 6	10
4.7	Пример 7	10
4.8	Пример 8	10
4.9	Пример 9	10
4.10	Пример 10	10
4.11	Пример 11	11
4.12	Пример 12	11
4.13	Пример 13	11
4.14	Пример 14	11
4.15	Копирование файла	11
4.16	Создание директории	12
4.17	Перемещение	12
4.18	Переименование	12
4.19	Создание и копирование	12
4.20	Создание каталога	12
4.21	Перемещение	12
4.22	Plans	12
4.23	australia	13
4.24	play	13
4.25	my_os	13
4.26	feathers	13
4.27	Просмотр	14
4.28	Копирование	14
4.29	Перемещение	14
4.30	Копирование	14
4.31	Перемещение и название	14
4.32	Лишение прав	15
4.33	Право на чтение	15
4.34	Лишение прав	15
4.35	Переход	15
4.36	Право на выполнение	15
4.37	mount	16

4.38 fsck	16
4.39 mkfs	17
4.40 kill	17

1 Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы

2 Задание

1. Выполнить примеры
2. Работа с файлами
3. Изменение прав доступа
4. Работа с файлами и правами доступа
5. Изучение команд с помощью man

3 Теоретическое введение

Для создания текстового файла можно использовать команду `touch`. Для просмотра файлов небольшого размера можно использовать команду `cat`. Для просмотра файлов постранично удобнее использовать команду `less`. Команда `cp` используется для копирования файлов и каталогов. Команды `mv` и `mkdir` предназначены для перемещения и переименования файлов и каталогов. Каждый файл или каталог имеет права доступа (табл. 5.1). В сведениях о файле или каталоге указываются: – тип файла (символ (-) обозначает файл, а символ (d) – каталог); – права для владельца файла (r – разрешено чтение, w – разрешена запись, x – разрешено выполнение, - – право доступа отсутствует); – права для членов группы (r – разрешено чтение, w – разрешена запись, x – разрешено выполнение, - – право доступа отсутствует); – права для всех остальных (r – разрешено чтение, w – разрешена запись, x – разрешено выполнение, - – право доступа отсутствует). Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора. Файловая система в Linux состоит из файлов и каталогов. Каждому физическому носителю соответствует своя файловая система. Существует несколько типов файловых систем. Перечислим наиболее часто встречающиеся типы: – ext2fs (second extended filesystem); – ext3fs (third extended file system); – ext4 (fourth extended file system); – ReiserFS; – xfs; – fat (file allocation table); – ntfs (new technology file system). В данном случае указаны имена устройств, названия соответствующих им точек монтирования (путь), тип файловой системы и параметрами монтирования. В контексте команды `mount`

устройство — специальный файл устройства, с помощью которого операционная система получает доступ к аппаратному устройству. Файлы устройств обычно располагаются в каталоге `/dev`, имеют сокращённые имена (например, `sdaN`, `sdbN` или `hdaN`, `hdbN`, где `N` — порядковый номер устройства, `sd` — устройства SCSI, `hd` — устройства MFM/IDE). Точка монтирования — каталог (путь к каталогу), к которому присоединяются файлы устройств. Для определения объёма свободного пространства на файловой системе можно воспользоваться командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования.

4 Выполнение лабораторной работы

4.1 Выполнение примеров из первой части описания лабораторной работы

Копирую файл в текущем каталоге (рис. 4.1).

```
eakiryanova@eakiryanova:~$ touch abc1
eakiryanova@eakiryanova:~$ touch april
eakiryanova@eakiryanova:~$ touch may
eakiryanova@eakiryanova:~$ cp abc1 april
eakiryanova@eakiryanova:~$ cp abc1 may
```

Рис. 4.1: Пример 1

Копирую несколько файлов в каталог (рис. 4.2).

```
eakiryanova@eakiryanova:~$ mkdir monthly
eakiryanova@eakiryanova:~$ cp april may monthly
```

Рис. 4.2: Пример 2

Копирую файлы в произвольном каталоге (рис. 4.3).

```
eakiryanova@eakiryanova:~$ cp monthly/may monthly/june
eakiryanova@eakiryanova:~$ ls monthly
april  june  may
```

Рис. 4.3: Пример 3

Копирую каталоги в текущем каталоге (рис. 4.4).

```
eakiryanova@eakiryanova:~$ mkdir monthly.00
eakiryanova@eakiryanova:~$ cp -r monthly monthly.00
```

Рис. 4.4: Пример 4

Копирую каталогов в произвольном каталоге (рис. 4.5).

```
eakiryanova@eakiryanova:~$ cp -r monthly.00 /tmp
```

Рис. 4.5: Пример 5

Переименование файлов в текущем каталоге (рис. 4.6).

```
eakiryanova@eakiryanova:~$ mv april july
```

Рис. 4.6: Пример 6

Перемещаю файлы в другой каталог (рис. 4.7).

```
eakiryanova@eakiryanova:~$ mv july monthly.00
eakiryanova@eakiryanova:~$ ls monthly.00
july  monthly
eakiryanova@eakiryanova:~$ ls monthly
april  june  may
```

Рис. 4.7: Пример 7

Переименование каталогов в текущем каталоге (рис. 4.8).

```
eakiryanova@eakiryanova:~$ mv monthly.00 monthly.01
```

Рис. 4.8: Пример 8

Перемещаю каталог в другой каталог (рис. 4.9).

```
eakiryanova@eakiryanova:~$ mkdir reports
eakiryanova@eakiryanova:~$ mv monthly.01 reports
```

Рис. 4.9: Пример 9

Переименование каталога, не являющегося текущим (рис. 4.10).

```
eakiryanova@eakiryanova:~$ mv reports/monthly.01 reports/monthly
```

Рис. 4.10: Пример 10

Создаю файл с правом выполнения для владельца (рис. 4.11).

```
eakiryanova@eakiryanova:~$ touch may
eakiryanova@eakiryanova:~$ ls -l may
-rw-r--r--. 1 eakiryanova eakiryanova 0 мар 29 10:43 may
eakiryanova@eakiryanova:~$ chmod u+x may
eakiryanova@eakiryanova:~$ ls -l may
-rwxr--r--. 1 eakiryanova eakiryanova 0 мар 29 10:43 may
```

Рис. 4.11: Пример 11

Лишаю владельца права на выполнение (рис. 4.12).

```
eakiryanova@eakiryanova:~$ chmod u-x may
eakiryanova@eakiryanova:~$ ls -l may
-rw-r--r--. 1 eakiryanova eakiryanova 0 мар 29 10:43 may
```

Рис. 4.12: Пример 12

Создаю каталог с запретом на чтение для членов группы и всех остальных пользователей (рис. 4.13).

```
eakiryanova@eakiryanova:~$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
eakiryanova@eakiryanova:~$ chmod g-r, o-r monthly
```

Рис. 4.13: Пример 13

Создаю файл с правом записи для членов группы (рис. 4.14).

```
eakiryanova@eakiryanova:~$ touch abc1
eakiryanova@eakiryanova:~$ chmod g+w abc1
```

Рис. 4.14: Пример 14

4.2 Работа с файлами

Копирую файл в домашний каталог и называю его equipment (рис. 4.15).

```
eakiryanova@eakiryanova:~$ cp /usr/include/sys/io.h ~/equipment
```

Рис. 4.15: Копирование файла

Создаю директорию (рис. 4.16).

```
eakiryanova@eakiryanova:~$ mkdir ~/ski.plases
```

Рис. 4.16: Создание директории

Перемещаю файл equipment в каталог ski.plases (рис. 4.17).

```
eakiryanova@eakiryanova:~$ mv ~/equipment ~/ski.plases/
```

Рис. 4.17: Перемещение

Переименовываю файл (рис. 4.10).

```
eakiryanova@eakiryanova:~$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
```

Рис. 4.18: Переименование

Создаю в домашнем каталоге файл и копирую его в каталог (рис. 4.19).

```
eakiryanova@eakiryanova:~$ touch ~/abc1  
eakiryanova@eakiryanova:~$ cp ~/abc1 ~/ski.plases/equiplist2
```

Рис. 4.19: Создание и копирование

Создаю каталог (рис. 4.20).

```
eakiryanova@eakiryanova:~$ mkdir ~/ski.plases/equipment
```

Рис. 4.20: Создание каталога

Перемещаю файлы в каталог (рис. 4.21).

```
eakiryanova@eakiryanova:~$ mv ~/ski.plases/equiplist ~/ski.plases/equipment/  
eakiryanova@eakiryanova:~$ mv ~/ski.plases/equiplist2  
mv: после '/home/eakiryanova/ski.plases/equiplist2' пропущен операнд, задающий целевой файл  
По команде «mv --help» можно получить дополнительную информацию.  
eakiryanova@eakiryanova:~$ mv ~/ski.plases/equiplist2 ~/ski.plases/equipment/
```

Рис. 4.21: Перемещение

Создаю, перемещаю каталог и называю его plans (рис. 4.22).

```
eakiryanova@eakiryanova:~$ mkdir ~/newdir  
eakiryanova@eakiryanova:~$ mv ~/newdir ~/ski.plases/plans
```

Рис. 4.22: Plans

4.3 Изменение прав доступа

Владелец имеет полный доступ, группа имеет право на чтение, остальные имеют право на чтение (рис. 4.23).

```
eakiryanova@eakiryanova:~$ mkdir australia  
eakiryanova@eakiryanova:~$ chmod 744 australia
```

Рис. 4.23: australia

Владелец имеет полный доступ, группа и остальные имеют право только на выполнение (рис. 4.24).

```
eakiryanova@eakiryanova:~$ mkdir play  
eakiryanova@eakiryanova:~$ chmod 711 play
```

Рис. 4.24: play

Владелец имеет право на чтение и выполнение, группа имеет право на чтение, остальные имеют право на чтение (рис. 4.25).

```
eakiryanova@eakiryanova:~$ touch my_os  
eakiryanova@eakiryanova:~$ chmod 544 my_os
```

Рис. 4.25: my_os

Владелец и группа имеют право на чтение и запись, остальные имеют право на чтение (рис. 4.26).

```
eakiryanova@eakiryanova:~$ touch feathers  
eakiryanova@eakiryanova:~$ chmod 664 feathers
```

Рис. 4.26: feathers

4.4 Работа с файлами и правами доступа

Просматриваю содержимое файла с помощью cat (рис. 4.27).

```

apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
tss:x:59:59:Account used for TPM access:/usr/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
geoclue:x:999:999:User for geoclue:/var/lib/geoclue:/sbin/nologin
usbmuxd:x:113:113:usbmuxd user:/sbin/nologin
systemd-oom:x:998:998:systemd Userspace OOM Killer:/usr/sbin/nologin
qemu:x:107:107:qemu user:/sbin/nologin
polkitd:x:114:114:User for polkitd:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/sbin/nologin
chrony:x:997:994:chrony system user:/var/lib/chrony:/sbin/nologin
dnsmasq:x:996:993:Dnsmasq DHCP and DNS server:/var/lib/dnsmasq:/usr/sbin/nologin
gluster:x:995:992:GlusterFS daemons:/run/gluster:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
pipewire:x:994:991:PipeWire System Daemon:/run/pipewire:/usr/sbin/nologin
unbound:x:993:990:Unbound DNS resolver:/var/lib/unbound:/sbin/nologin
nm-openconnect:x:992:989:NetworkManager user for OpenConnect:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
wsdd:x:991:988:Web Services Dynamic Discovery host daemon:/sbin/nologin
sssd:x:990:986:User for sssd:/run/sss:/sbin/nologin
openvpn:x:989:985:OpenVPN:/etc/openvpn:/sbin/nologin

```

Рис. 4.27: Просмотр

Копирую файл (рис. 4.28).

```
eakiryanova@eakiryanova:~$ cp ~/feathers ~/file.old
```

Рис. 4.28: Копирование

Перемещаю файл (рис. 4.29).

```
eakiryanova@eakiryanova:~$ mv ~/file.old ~/play/
```

Рис. 4.29: Перемещение

Копирую каталог в каталог (рис. 4.30).

```
eakiryanova@eakiryanova:~$ cp -r ~/play ~/fun/
```

Рис. 4.30: Копирование

Перемещаю каталога в каталог и называю его games (рис. 4.31).

```
eakiryanova@eakiryanova:~$ mv ~/fun ~/play/games
```

Рис. 4.31: Перемещение и название

Лишаю владельца права на чтение. При попытке просмотреть файл выйдет ошибка. Копирование пройдет успешно, так как для него не требуется право на чтение исходного файла (рис. 4.32).

```
eakiryanova@eakiryanova:~$ chmod u-r ~/feathers
```

Рис. 4.32: Лишение прав

Даю владельцу файла право на чтение (рис. 4.33).

```
eakiryanova@eakiryanova:~$ chmod u+r ~/feathers
```

Рис. 4.33: Право на чтение

Лишаю владельца каталога права на выполнение (рис. 4.34).

```
eakiryanova@eakiryanova:~$ chmod u-x ~/play
```

Рис. 4.34: Лишение прав

При переходе в каталог выдало ошибку (рис. 4.35).

```
eakiryanova@eakiryanova:~$ cd ~/play  
bash: cd: /home/eakiryanova/play: Отказано в доступе
```

Рис. 4.35: Переход

Даю владельцу каталога право на выполнение (рис. 4.36).

```
eakiryanova@eakiryanova:~$ chmod u+x ~/play
```

Рис. 4.36: Право на выполнение

4.5 Изучение команд с помощью man

Команда mount используется для подключения файловых систем к дереву каталогов. Она позволяет монтировать устройства и разделы, чтобы получить доступ к их содержимому (рис. 4.37).

```
eakiryanova@eakiryanova:/home/eakiryanova — man mount
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
```

Рис. 4.37: mount

Команда `fsck` проверяет и восстанавливает целостность файловых систем. Она используется для обнаружения и исправления ошибок на файловых системах (рис. 4.38).

```
eakiryanova@eakiryanova:/home/eakiryanova — man fsck
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 a 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
```

Рис. 4.38: fsck

Команда `mkfs` используется для создания файловой системы на разделе или устройстве. Эта команда форматирует устройство, подготавливая его для хранения данных (рис. 4.39).


```
eakiryanova@eakiryanova:/home/eakiryanova — man mkfs
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.

OPTIONS
    -t, --type type
        Specify the type of filesystem to be built. If not specified, the default
        filesystem type (currently ext2) is used.
```

Рис. 4.39: mkfs

Команда kill используется для отправки сигналов процессам. Наиболее часто используется для завершения процессов, но также может использоваться для отправки других сигналов (рис. 4.40).

```
eakiryanova@eakiryanova:/home/eakiryanova — man kill
KILL(1) User Commands KILL(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--]
    pid|name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified processes or process
    groups.

    If no signal is specified, the TERM signal is sent. The default action for this signal
    is to terminate the process. This signal should be used in preference to the KILL
    signal (number 9), since a process may install a handler for the TERM signal in order
    to perform clean-up steps before terminating in an orderly fashion. If a process does
    not terminate after a TERM signal has been sent, then the KILL signal may be used; be
    aware that the latter signal cannot be caught, and so does not give the target process
    the opportunity to perform any clean-up before terminating.

    Most modern shells have a builtin kill command, with a usage rather similar to that of
    the command described here. The --all, --pid, and --queue options, and the possibility
    to specify processes by command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking is still performed.
```

Рис. 4.40: kill

5 Выводы

В ходе выполнения 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

Список литературы

1.Операционные системы