

Лабораторная работа № 7

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Мальянц Виктория Кареновна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
3.1	Выполнение примеров, приведенных в первой части описания лабораторной работы	7
3.2	Создание, перемещение и переименовывание каталогов и файлов	8
3.3	Работа с правами доступа	8
3.4	Выполнение упражнений	9
3.5	Работа с командами mount, fsck, mkfs, kill	10
4	Выводы	13
5	Контрольные вопросы	14
	Список литературы	16

Список иллюстраций

3.1	Выполнение примеров из первой части описания лабораторной работы	7
3.2	Выполнение примеров из первой части описания лабораторной работы	8
3.3	Применение различных команд	8
3.4	Создание каталогов australia и play	9
3.5	Создание файлов my_os и feathers	9
3.6	Присваивание выделенных прав доступа	9
3.7	Просмотр содержимого файла /etc/password	10
3.8	Копирование и перемещение файлов	10
3.9	Копирование, перемещение, задание прав доступа и просмотр каталогов и файлов	10
3.10	Копирование и задание прав доступа каталогов и файлов	10
3.11	Просмотр map по команде mount	11
3.12	Просмотр map по команде fsck	11
3.13	Просмотр map по команде mkfs	12
3.14	Просмотр map по команде kill	12

Список таблиц

1 Цель работы

Ознакомление с файловой системой Linux, ее 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.

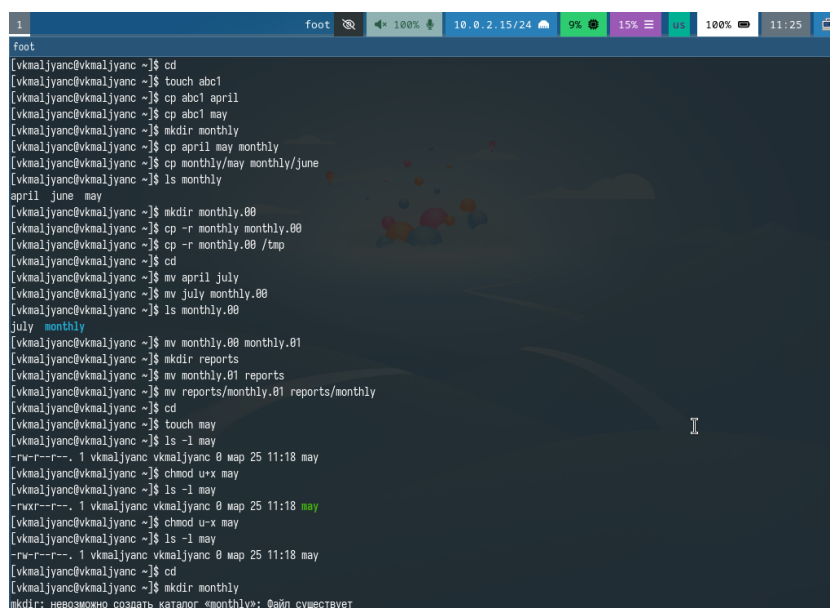
2 Задание

1. Выполнение примеров, приведенных в первой части описания лабораторной работы
2. Создание, перемещение и переименовывание каталогов и файлов
3. Работа с правами доступа
4. Выполнение упражнений
5. Работа с командами `mount`, `fsck`, `mkfs`, `kill`
6. Контрольные вопросы

3 Выполнение лабораторной работы

3.1 Выполнение примеров, приведенных в первой части описания лабораторной работы

Выполняю примеры из первой части описания лабораторной работы (рис. 3.1) (рис. 3.2).



```
1 foot 10.0.2.15/24 9% 15% 100% 11:25
foot
[vkmaljyanc@vkmaljyanc ~]$ cd
[vkmaljyanc@vkmaljyanc ~]$ touch abc1
[vkmaljyanc@vkmaljyanc ~]$ cp abc1 april
[vkmaljyanc@vkmaljyanc ~]$ cp abc1 may
[vkmaljyanc@vkmaljyanc ~]$ mkdir monthly
[vkmaljyanc@vkmaljyanc ~]$ cp april may monthly
[vkmaljyanc@vkmaljyanc ~]$ cp monthly/may monthly/june
[vkmaljyanc@vkmaljyanc ~]$ ls monthly
april  june  may
[vkmaljyanc@vkmaljyanc ~]$ mkdir monthly.00
[vkmaljyanc@vkmaljyanc ~]$ cp -r monthly monthly.00
[vkmaljyanc@vkmaljyanc ~]$ cp -r monthly.00 /tmp
[vkmaljyanc@vkmaljyanc ~]$ cd
[vkmaljyanc@vkmaljyanc ~]$ mv april july
[vkmaljyanc@vkmaljyanc ~]$ mv july monthly.00
[vkmaljyanc@vkmaljyanc ~]$ ls monthly.00
july  monthly
[vkmaljyanc@vkmaljyanc ~]$ mv monthly.00 monthly.01
[vkmaljyanc@vkmaljyanc ~]$ mkdir reports
[vkmaljyanc@vkmaljyanc ~]$ mv monthly.01 reports
[vkmaljyanc@vkmaljyanc ~]$ mv reports/monthly.01 reports/monthly
[vkmaljyanc@vkmaljyanc ~]$ cd
[vkmaljyanc@vkmaljyanc ~]$ touch may
[vkmaljyanc@vkmaljyanc ~]$ ls -l may
-rw-r--r--. 1 vkmaljyanc vkmaljyanc 0 map 25 11:18 may
[vkmaljyanc@vkmaljyanc ~]$ chmod u+x may
[vkmaljyanc@vkmaljyanc ~]$ ls -l may
-rwxr--r--. 1 vkmaljyanc vkmaljyanc 0 map 25 11:18 may
[vkmaljyanc@vkmaljyanc ~]$ chmod u-x may
[vkmaljyanc@vkmaljyanc ~]$ ls -l may
-rw-r--r--. 1 vkmaljyanc vkmaljyanc 0 map 25 11:18 may
[vkmaljyanc@vkmaljyanc ~]$ cd
[vkmaljyanc@vkmaljyanc ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: файл существует
```

Рис. 3.1: Выполнение примеров из первой части описания лабораторной работы

```

[vkmaljyanc@vkmaljyanc ~]$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
[vkmaljyanc@vkmaljyanc ~]$ cd
[vkmaljyanc@vkmaljyanc ~]$ touch abc1
[vkmaljyanc@vkmaljyanc ~]$ chmod g+w abc1
[vkmaljyanc@vkmaljyanc ~]$ fsck /dev/sda1
fsck from util-linux 2.40.4
fsck.fat 4.2 (2021-01-31)
open: Permission denied
[vkmaljyanc@vkmaljyanc ~]$

```

Рис. 3.2: Выполнение примеров из первой части описания лабораторной работы

3.2 Создание, перемещение и переименовывание каталогов и файлов

Копирую файл `/usr/include/sys/io.h` в домашний каталог и называю его `equipment`. В домашнем каталоге создаю директорию `~/ski.places`. Перемещаю файл `equipment` в каталог `~/ski.places`. Переименовываю файл `~/ski.places/equipment` в `~/ski.places/equiplist`. Создаю в домашнем каталоге файл `abc1` и копирую его в каталог `~/ski.places`, называю его `equiplist2`. Создаю каталог с именем `equipment` в каталоге `~/ski.places`. Перемещаю файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`. Создаю и перемещаю каталог `~/newdir` в каталог `~/ski.places` и называю его `plans` (рис. 3.3).

```

[vkmaljyanc@vkmaljyanc ~]$ cp /usr/include/sys/io.h ~/equipment
[vkmaljyanc@vkmaljyanc ~]$ mkdir ~/ski.places
[vkmaljyanc@vkmaljyanc ~]$ mv ~/equipment ~/ski.places
[vkmaljyanc@vkmaljyanc ~]$ mv ~/ski.places/equipment ~/ski.places/equiplist
[vkmaljyanc@vkmaljyanc ~]$ touch abc1
[vkmaljyanc@vkmaljyanc ~]$ cp abc1 ~/ski.places/equiplist2
[vkmaljyanc@vkmaljyanc ~]$ mkdir ~/ski.places/equipment
[vkmaljyanc@vkmaljyanc ~]$ mv ~/ski.places/equiplist ~/ski.places/equipment
[vkmaljyanc@vkmaljyanc ~]$ mv ~/ski.places/equiplist2 ~/ski.places/equipment
[vkmaljyanc@vkmaljyanc ~]$ mkdir ~/newdir
mkdir: невозможно создать каталог «/home/vkmaljyanc/newdir»: файл существует
[vkmaljyanc@vkmaljyanc ~]$ mv ~/newdir ~/ski.places/plans
[vkmaljyanc@vkmaljyanc ~]$

```

Рис. 3.3: Применение различных команд

3.3 Работа с правами доступа

Создаю каталоги `australia` и `play` и файлы `my_os` и `feathers`. Присваиваю этим каталогам и файлам выделенные права доступа (рис. 3.4) (рис. 3.5) (рис. 3.6).


```
[vkmajjanc@vkmajjanc ~]$ mkdir australia
[vkmajjanc@vkmajjanc ~]$ mkdir play
```

Рис. 3.4: Создание каталогов australia и play

```
[vkmajjanc@vkmajjanc ~]$ touch play
[vkmajjanc@vkmajjanc ~]$ touch my_os
```

Рис. 3.5: Создание файлов my_os и feathers

```
[vkmajjanc@vkmajjanc ~]$ chmod 744 australia
[vkmajjanc@vkmajjanc ~]$ chmod 711 play
[vkmajjanc@vkmajjanc ~]$ chmod 544 my_os
[vkmajjanc@vkmajjanc ~]$ chmod 664 feathers
[vkmajjanc@vkmajjanc ~]$ ls -l
иторо 9780
-rw-rw-r--. 1 vkmajjanc vkmajjanc      0 мар 25 11:31 abc1
drwxr--r--. 1 vkmajjanc vkmajjanc      0 мар 25 11:42 australia
drwxr-xr-x. 1 vkmajjanc vkmajjanc    14 мар 11 19:34 bin
drwxr-xr-x. 1 vkmajjanc vkmajjanc   290 мар 21 12:49 blog
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 мар 21 11:39 Desktop
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 мар 11 21:20 Documents
drwxr-xr-x. 1 vkmajjanc vkmajjanc   310 мар 22 16:08 Downloads
-rw-rw-r--. 1 vkmajjanc vkmajjanc      0 мар 25 11:41 feathers
drwxr-xr-x. 1 vkmajjanc vkmajjanc    74 мар 7 21:48 git-extended
-rw-rw-r--. 1 vkmajjanc vkmajjanc      0 мар 25 11:18 may
drwxr-xr-x. 1 vkmajjanc vkmajjanc    24 мар 25 11:12 monthly
-rw-rw-r--. 1 vkmajjanc vkmajjanc      0 мар 25 11:41 my_os
drwx-x-x. 1 vkmajjanc vkmajjanc      0 мар 25 11:42 play
-rw-rw-r--. 1 vkmajjanc vkmajjanc 6615080 мар 16 18:05 presentation.html
-rw-rw-r--. 1 vkmajjanc vkmajjanc   8658 мар 16 18:03 presentation.md
-rw-rw-r--. 1 vkmajjanc vkmajjanc 3297825 мар 16 18:05 presentation.pdf
drwxr-xr-x. 1 vkmajjanc vkmajjanc    14 мар 25 11:18 reports
drwxr-xr-x. 1 vkmajjanc vkmajjanc    28 мар 25 11:37 ski.places
drwxr-xr-x. 1 vkmajjanc vkmajjanc    10 мар 16 15:39 work
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 фев 24 14:14 Видео
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 фев 24 14:14 Документы
drwxr-xr-x. 1 vkmajjanc vkmajjanc   134 мар 8 16:05 Загрузки
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 мар 1 21:44 Изображения
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 фев 24 14:14 Музыка
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 фев 24 14:14 Общедоступные
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 мар 7 22:49 'Рабочий стол'
drwxr-xr-x. 1 vkmajjanc vkmajjanc      0 фев 24 14:14 Шаблоны
[vkmajjanc@vkmajjanc ~]$
```

Рис. 3.6: Присваивание выделенных прав доступа

3.4 Выполнение упражнений

Просматриваю содержимое файла /etc/passwd. Копирую файл ~/feathers в файл ~/file.old. Перемещаю файл ~/file.old в каталог ~/play. Копирую каталог ~/play в каталог ~/fun. Перемещаю каталог ~/fun в каталог ~/play и называю его games. Лишаю владельца файла ~/feathers права на чтение. Пытаюсь просмотреть файл ~/feathers командой cat, отказано в доступе. Даю владельцу файла ~/feathers право на чтение. Лишаю владельца каталога ~/play права на выполнение. Перехожу в каталог ~/play, отказано в доступе. Даю владельцу каталога ~/play право на выполнение (рис. 3.7) (рис. 3.8) (рис. 3.9) (рис. 3.10).

```
[vkmajjanc@vkmajjanc ~]$ cat /etc/passwd
root:x:0:0:Super User:/root:/bin/bash
bin:x:1:1:bin:/bin:/usr/sbin/nologin
daemon:x:2:2:daemon:/sbin:/usr/sbin/nologin
adm:x:3:4:adm:/var/adm:/usr/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/usr/sbin/nologin
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:/var/spool/mail:/usr/sbin/nologin
operator:x:11:0:operator:/root:/usr/sbin/nologin
games:x:12:100:games:/usr/games:/usr/sbin/nologin
ftp:x:14:50:FTP User:/var/ftp:/usr/sbin/nologin
nobody:x:65534:65534:Kernel Overflow User:/usr/sbin/nologin
dbus:x:81:81:System Message Bus:/usr/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
systemd-oom:x:998:998:systemd Userspace OOM Killer:/usr/sbin/nologin
polkitd:x:114:114>User for polkitd:/sbin/nologin
sstpc:x:997:995:Secure Socket Tunneling Protocol (SSTP) Client:/var/run/sstpc:/sbin/nologin
rtkit:x:172:172:RealtimeKit:/sbin/nologin
chrony:x:996:994:chrony system user:/var/lib/chrony:/sbin/nologin
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
pipewire:x:995:993:Pipewire System Daemon:/run/pipewire:/usr/sbin/nologin
sssd:x:994:992>User for sssd:/run/sss:/sbin/nologin
unbound:x:993:991:Unbound DNS resolver:/var/lib/unbound:/sbin/nologin
nm-openconnect:x:992:990:NetworkManager user for OpenConnect:/sbin/nologin
vsdd:x:991:989:Web Services Dynamic Discovery host daemon:/sbin/nologin
openvpn:x:990:988:OpenVPN:/etc/openvpn:/sbin/nologin
nm-openvpn:x:989:987:Default user for running openvpn spawned by NetworkManager:/sbin/nologin
colord:x:988:986>User for colord:/var/lib/colord:/sbin/nologin
```

Рис. 3.7: Просмотр содержимого файла /etc/password

```
[vkmajjanc@vkmajjanc ~]$ cp ~/feathers ~/file.old
[vkmajjanc@vkmajjanc ~]$ mv ~/file.old ~/play
```

Рис. 3.8: Копирование и перемещение файлов

```
[vkmajjanc@vkmajjanc ~]$ cp -r ~/play ~/fun
[vkmajjanc@vkmajjanc ~]$ mv ~/fun ~/play/games
[vkmajjanc@vkmajjanc ~]$ chmod u-r feathers
[vkmajjanc@vkmajjanc ~]$ cat ~/feathers
cat: /home/vkmajjanc/feathers: Отказано в доступе
```

Рис. 3.9: Копирование, перемещение, задание прав доступа и просмотр каталогов и файлов

```
[vkmajjanc@vkmajjanc ~]$ cp ~/feathers ~/play
cp: невозможно открыть '/home/vkmajjanc/feathers' для чтения: Отказано в доступе
[vkmajjanc@vkmajjanc ~]$ chmod u+r feathers
[vkmajjanc@vkmajjanc ~]$ chmod u-x play
[vkmajjanc@vkmajjanc ~]$ cd ~/play
bash: cd: /home/vkmajjanc/play: Отказано в доступе
[vkmajjanc@vkmajjanc ~]$ chmod u+x play
[vkmajjanc@vkmajjanc ~]$
```

Рис. 3.10: Копирование и задание прав доступа каталогов и файлов

3.5 Работа с командами mount, fsck, mkfs, kill

Просматриваю man по команде mount (рис. 3.11).

```
1 foot 10.0.2.15/24 17% 17% 100% 11:59
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|rsaved|rsaved|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 the filesystem found on some device to the big file tree. Conversely, the umount(8) command will detach it again. The filesystem is used to control how data is stored on the device or provided in a virtual way by network or other services.

  The standard form of the mount command is:

      mount -t type device dir

  This tells the kernel to attach the filesystem found on device (which is of type type) at the directory dir. The option -t type is optional. The mount command is usually able to detect a filesystem. The root permissions are necessary to mount a filesystem by default. See section "Non-superuser mounts" below for more details. The previous contents (if any) and owner and mode of dir become invisible, and as long as this filesystem remains mounted, the pathname dir refers to the root of the filesystem on device.

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

Рис. 3.11: Просмотр man по команде mount

Просматриваю man по команде fsck (рис. 3.12).

```
1 foot 10.0.2.15/24 9% 17% 100% 11:59
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/hda1, /dev/sdb2), a mount point (e.g., /, /usr, /home), or a filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a83-98b8-bfc24857f7bd 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

  1      Filesystem errors corrected

  2      System should be rebooted

  4      Filesystem errors left uncorrected

  8      Operational error

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

Рис. 3.12: Просмотр man по команде fsck

Просматриваю man по команде mkfs (рис. 3.13).

```
1 foot 100% 10.0.2.15/24 9% 17% 100% 12:01
Foot
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.

  fs-options
    Filesystem-specific options to be passed to the real filesystem builder.

  -V, --verbose
    Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.

  -h, --help
    Display help text and exit.

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

Рис. 3.13: Просмотр ман по команде mkfs

Просматриваю ман по команде kill (рис. 3.14 [1]).

```
1 foot 100% 10.0.2.15/24 8% 17% 100% 12:01
Foot
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.

ARGUMENTS
  The list of processes to be signaled can be a mixture of names and PIDs.

  pid
    Each pid can be expressed in one of the following ways:

    n
      where n is larger than 0. The process with PID n is signaled.

    0
```

Рис. 3.14: Просмотр ман по команде kill

4 Выводы

Я ознакомилась с файловой системой Linux, ее 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.

5 Контрольные вопросы

1. ext4 - файловая система, используемая по умолчанию в большинстве дистрибутивов Linux, включая Fedora. Высокая производительность, надежность, поддержка больших файлов и разделов.
2. /root - корневой каталог /bin - содержит основные исполняемые файлы /boot - содержит файлы, необходимые для загрузки операционной системы /dev - содержит файлы устройств, представляющие аппаратные устройства /etc - содержит конфигурационные файлы системы и приложений /home - содержит домашние каталоги пользователей /lib - содержит общие библиотеки, необходимые для работы исполняемых файлов /media - используется для автоматического монтирования съемных носителей /mnt - используется как временная точка монтирования для файловых систем /opt - предназначен для установки дополнительных программных пакетов /proc - специальная виртуальная файловая система, которая содержит информацию о процессах, запущенных в системе /run - временная файловая система, используемая для хранения данных, необходимых для работы системы после загрузки /sbin - содержит системные исполняемые файлы, предназначенные для администрирования системы /srv - содержит данные, предоставляемые системой как сервисы /sys - специальная виртуальная файловая система, которая предоставляет информацию о ядре и устройствах /tmp - каталог для хранения временных файлов /usr - содержит основные ресурсы системы, такие как исполняемые файлы, библиотеки, документацию и исходный код /var - содержит файлы, содержимое которых часто

- меняется, такие как логи, базы данных, временные файлы, очереди печати
3. С помощью команды `mount`
 4. Основные причины нарушения целостности файловой системы: программные ошибки, аппаратные сбои, неправильное выключение системы, сбои питания, вирусы и вредоносные ПО. Методы устранения повреждений файловой системы: применение команды `fsck`, восстановление из резервной копии, профессиональное восстановление данных
 5. С помощью команды `mkfs`
 6. `cat` - выводит содержимое файла в стандартный поток вывода, `less` - позволяет просматривать большие файлы постранично, `more` - подобен `less`, но имеет меньше возможностей навигации, `head` - выводит первые `n` строк файла (по умолчанию 10), `tail` - выводит последние `n` строк файла (по умолчанию 10)
 7. `cp` - копирование файлов, каталогов, сохранение атрибутов файлов, копирование с подтверждением, создание символических ссылок, копирование с сохранением структуры каталогов
 8. `mv` - перемещение и переименование файлов и каталогов, перемещение с подтверждением, перемещение, если нет прав перезаписи
 9. Права доступа - определяют, какие операции могут выполнять пользователи и группы над файлами и каталогами. Существуют три типа прав доступа: чтение, запись, выполнение. Изменить права доступа можно с помощью разных форм записи: двоичной, восьмеричной и символьной.

Список литературы

1. Лабораторная работа № 7.