

# **Лабораторная работа №7**

*Дисциплина: Операционные системы*

Долгаев Евгений НММбд-01-24

# Содержание

|          |  |           |
|----------|--|-----------|
| <b>1</b> | <b>Цель работы</b>   | <b>5</b>  |
| <b>2</b> | <b>Задание</b>   | <b>6</b>  |
| <b>3</b> | <b>Выполнение лабораторной работы</b>                        | <b>7</b>  |
| 3.1      | Краткая характеристика команд mount, fsck, mkfs и kill . . . | 11        |
| 3.1.1    | mount . . . . .  | 11        |
| 3.1.2    | fsck . . . . .   | 12        |
| 3.1.3    | mkfs . . . . .   | 13        |
| 3.1.4    | kill . . . . .   | 13        |
| <b>4</b> | <b>Ответы на контрольные вопросы</b>                         | <b>15</b> |
| 4.0.1    | Вопрос 1 . . . . .   | 15        |
| 4.0.2    | Вопрос 2 . . . . .   | 15        |
| 4.0.3    | Вопрос 3 . . . . .   | 18        |
| 4.0.4    | Вопрос 4 . . . . .   | 18        |
| 4.0.5    | Вопрос 5 . . . . .   | 19        |
| 4.0.6    | Вопрос 6 . . . . .   | 19        |
| 4.0.7    | Вопрос 7 . . . . .   | 19        |
| 4.0.8    | Вопрос 8 . . . . .   | 19        |
| 4.0.9    | Вопрос 9 . . . . .   | 19        |
| <b>5</b> | <b>Выводы</b>  | <b>21</b> |
|          | <b>Список литературы</b>                                     | <b>22</b> |

# Список иллюстраций

|      |                     |    |
|------|---------------------|----|
| 3.1  | Задание 1 . . . . . | 7  |
| 3.2  | Задание 1 . . . . . | 8  |
| 3.3  | Задание 1 . . . . . | 8  |
| 3.4  | Задание 1 . . . . . | 8  |
| 3.5  | Задание 1 . . . . . | 8  |
| 3.6  | Задание 2 . . . . . | 9  |
| 3.7  | Задание 3 . . . . . | 10 |
| 3.8  | Задание 3 . . . . . | 10 |
| 3.9  | Задание 4 . . . . . | 11 |
| 3.10 | mount . . . . .     | 12 |
| 3.11 | fsck . . . . .      | 12 |
| 3.12 | mkfs . . . . .      | 13 |
| 3.13 | kill . . . . .      | 14 |
| 4.1  | fs . . . . .        | 16 |

# Список таблиц

# 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 все примеры, приведённые в первой части описания лабораторной работы.
- 2) Упражнения.
- 3) Определить опции команды `chmod`, необходимые для того, чтобы присвоить файлам выделенные права доступа.
- 4) Упражнения.
- 5) Прочитать ман по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризовать, приведя примеры.

### 3 Выполнение лабораторной работы

На следующих скриншотах приведены результаты выполнения команд, которые давались как примеры в теоретической части содержания лабораторной работы (рис. 3.1, 3.2, 3.3, 3.4, 3.5).

```
root@esdolghev:~# cd
root@esdolghev:~# touch abc1
root@esdolghev:~# cd abc1 april
-bash: cd: cannot access april: No such file or directory
root@esdolghev:~# cp abc1 april
root@esdolghev:~# cp abc1 may
root@esdolghev:~# mkdir monthly
root@esdolghev:~# cp april may monthly
root@esdolghev:~# cp monthly/may monthly/june
root@esdolghev:~# ls monthly
april  june  may
root@esdolghev:~# mkdir monthly.00
-bash: mkdir: cannot create directory 'monthly.00'
root@esdolghev:~# mkdir monthly.00
root@esdolghev:~# cp -r monthly.00 /tmp
root@esdolghev:~# cd
root@esdolghev:~# mv april july
root@esdolghev:~# mv july monthly.00
root@esdolghev:~# ls monthly.00
july
root@esdolghev:~# mv -i monthly.00 monthly.01
root@esdolghev:~# mkdir reports
root@esdolghev:~# mv monthly.01 reports
root@esdolghev:~# mv reports/monthly.01 reports/monthly
root@esdolghev:~# cd
root@esdolghev:~# touch may
root@esdolghev:~# ls -l may
-rw-r--r-- 1 root root 0 map 29 12:33 may
root@esdolghev:~# chmod u-x may
root@esdolghev:~# ls -l may
-rw-r--r-- 1 root root 0 map 29 12:33 may
root@esdolghev:~# cd
root@esdolghev:~# mkdir monthly
mkdir: невозможно создать каталог «monthly». Файл существует
root@esdolghev:~# chmod g-r, g-o monthly
chmod: несправ режим: «g-r.»
По команде «chmod --help» можно получить дополнительную информацию.
root@esdolghev:~# chmod g-r, o-r monthly
root@esdolghev:~# chmod g-r, o-r monthly
root@esdolghev:~# cd
root@esdolghev:~# touch abc1
root@esdolghev:~# chmod g-w abc1
root@esdolghev:~#
```

Рис. 3.1: Задание 1

```

root@esdolgav:~# mount
/dev/vda3 on / type btrfs (rw,relatime,seclabel,compress=zstd:1,discard=async,space_cache=v2,subvolid=257,subvol=/root)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=4096k,nr_inodes=245388,mode=755,inode64)
tmpfs on /dev/pts type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,nodev,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nodelegat,memory=recursiveprot)
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
efivarfs on /sys/firmware/efi/efivars type efivarfs (rw,nosuid,nodev,noexec,relatime)
bpf on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=99972k,nr_inodes=819200,mode=755,inode64)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,nodev,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=35,pgrp=1,timeout=0,nrproto=5,naxproto=5,direct,pipe_ino=4764)
tunefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugetlbfs type hugetlbfs (rw,nosuid,nodev,relatime,seclabel,pagesize=2M)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tmpfs on /run/credentials/system-network-generator.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/system-journald.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/system-udev-load-credentials.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/systemd-sysctl.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/systemd-tmpfiles-setup-dev-early.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/systemd-tmpfiles-setup-dev.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
/dev/vda3 on /home type btrfs (rw,relatime,seclabel,compress=zstd:1,discard=async,space_cache=v2,subvolid=256,subvol=/home)
tmpfs on /tmp type tmpfs (rw,nosuid,nodev,seclabel,nr_inodes=184576,inode64)
/dev/vda2 on /boot type ext4 (rw,relatime,seclabel)
/dev/vda1 on /boot/efi type vfat (rw,relatime,fmask=0077,dmask=0077,codepage=437,iocharset=utf8,shortname=winnt,errors=remount-ro)
tmpfs on /run/credentials/systemd-tmpfiles-setup.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/systemd-vconsole-setup.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /run/credentials/systemd-resolved.service type tmpfs (ro,nosuid,nodev,noexec,relatime,nosymlinks,seclabel,size=1824k,nr_inodes=1824,mode=700,inode64,noswap)
tmpfs on /var/lib/flatpak/exports type tmpfs (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=199784k,nr_inodes=49946,mode=700,uid=1000,gid=1000,inode64)
pstore on /run/user/1000 type pstore (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
tmpfs on /run/user/0 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=199784k,nr_inodes=49946,mode=700,uid=0,inode64)
root@esdolgav:~#

```

Рис. 3.2: Задание 1

```

root@esdolgav:~# cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Sun Feb 16 04:34:49 2025
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=2ec7fec0-2962-4485-8217-6ec4322e8e59 / btrfs subvol=root,compress=zstd:1 0 0
UUID=88116d0f-3bc9-4dd1-ac43-31c2878edf0e /boot ext4 defaults 1 2
UUID=0C56-C8AB /boot/efi vfat umask=0077,shortname=winnt 0 2
UUID=2ec7fec0-2962-4485-8217-6ec4322e8e59 /home btrfs subvol=home,compress=zstd:1 0 0
root@esdolgav:~#
[0] 0: sudo*

```

Рис. 3.3: Задание 1

```

root@esdolgav:~# df
Файловая система 1K-блоков Использовано Доступно Использовано% Смониторовано в
/dev/vda3 61249536 10480756 49651564 18% /
devtmpfs 4096 0 4096 0% /dev
tmpfs 998924 7348 991576 1% /dev/shm
efivarfs 256 21 231 9% /sys/firmware/efi/efivars
tmpfs 399572 1176 398396 1% /run
tmpfs 1024 0 1024 0% /run/credentials/systemd-network-generator.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-journald.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-udev-load-credentials.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-sysctl.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-tmpfiles-setup-dev-early.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-tmpfiles-setup-dev.service
/dev/vda3 61249536 10480756 49651564 18% /home
tmpfs 998924 4 998920 1% /tmp
/dev/vda2 996780 299952 628016 33% /boot
/dev/vda1 613160 19808 593352 4% /boot/efi
tmpfs 1024 0 1024 0% /run/credentials/systemd-tmpfiles-setup.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-vconsole-setup.service
tmpfs 1024 0 1024 0% /run/credentials/systemd-resolved.service
tmpfs 199784 96 199688 1% /run/user/1000
tmpfs 199784 64 199720 1% /run/user/0
root@esdolgav:~#
[0] 0: sudo*

```

Рис. 3.4: Задание 1

```

root@esdolgav:~# fsck /dev/sda1
fsck from util-linux 2.40.4
e2fsck 1.47.1 (20-May-2024)
fsck.ext2: Нет такого файла или каталога while trying to open /dev/sda1
Possibly non-existent device?
root@esdolgav:~#

```

Рис. 3.5: Задание 1



Далее идут упражнения, направленные на отработку навыков работы с файловой системой (рис. 3.6).

```
root@esdolgaev:~# cp /usr/include/sys/io.h /home/esdolgaev/equipment.h
root@esdolgaev:~# mkdir sky.plases
root@esdolgaev:~# mv equipment.h ski.plases
mv: не удалось выполнить stat для 'equipment.h': Нет такого файла или каталога
root@esdolgaev:~# mv /home/esdolgaev/equipment.h /home/esdolgaev/ski.plases

root@esdolgaev:~# mv /home/esdolgaev/sky.plases/equipment.h /home/esdolgaev/ski.plases/equiplist.h

mv: не удалось выполнить stat для '/home/esdolgaev/sky.plases/equipment.h': Нет такого файла или каталога
root@esdolgaev:~# mc

root@esdolgaev:~# cd sky.plases
root@esdolgaev:~/sky.plases# touch equipment.h
root@esdolgaev:~/sky.plases# cd
root@esdolgaev:~# mv /home/esdolgaev/sky.plases/equipment.h /home/esdolgaev/ski.plases/equiplist.h

mv: не удалось выполнить stat для '/home/esdolgaev/sky.plases/equipment.h': Нет такого файла или каталога
root@esdolgaev:~# ls sky.plases
equipment.h
root@esdolgaev:~# mv sky.plases/equipment.h sky.plases/equiplist.h
root@esdolgaev:~# ls sky.plases
equiplist.h
root@esdolgaev:~#
Display all 378 possibilities? (y or n)
root@esdolgaev:~# touch abc1
root@esdolgaev:~# mv abc1 sky.plases/equiplist2.h
root@esdolgaev:~# cd sky.plases
root@esdolgaev:~/sky.plases# mkdir equipment
root@esdolgaev:~/sky.plases# mv equiplist.h equipment
root@esdolgaev:~/sky.plases# mv equiplist2.h equipment
root@esdolgaev:~/sky.plases# ls equipment
equiplist2.h  equiplist.h
root@esdolgaev:~/sky.plases# cd
root@esdolgaev:~# mkdir newdir
root@esdolgaev:~# mv newdir sky.plases/plans
root@esdolgaev:~# ls sky.plases
equipment  plans
root@esdolgaev:~#
```

Рис. 3.6: Задание 2

Задание 3 позволяет попробовать в действии команду `chmod` (рис. 3.7, 3.8).

```

root
australia feathers my_os play
root@esdolghev:~/ex3# ls -l
иторо 0
-rw-r--r-- 1 root root 0 map 29 12:54 australia
-rw-r--r-- 1 root root 0 map 29 12:54 feathers
-rw-r--r-- 1 root root 0 map 29 12:54 my_os
-rw-r--r-- 1 root root 0 map 29 12:54 play
root@esdolghev:~/ex3# chmod u+x australia
root@esdolghev:~/ex3# ls -l australia
-rwxr--r-- 1 root root 0 map 29 12:54 australia
root@esdolghev:~/ex3# chmod u+d australia
chmod: неверный режим: «u+d»
По команде «chmod --help» можно получить дополнительную информацию.
root@esdolghev:~/ex3# rm australia
root@esdolghev:~/ex3# rm play
root@esdolghev:~/ex3# mkdir australia
root@esdolghev:~/ex3# mkdir play
root@esdolghev:~/ex3# chmod u+x australia
root@esdolghev:~/ex3# ls -l australia
ls: невозможно получить доступ к 'australia': Нет такого файла или каталога
root@esdolghev:~/ex3# ls -l australia
иторо 0
root@esdolghev:~/ex3# ls -l
иторо 0
drwxr-xr-x 1 root root 0 map 29 13:03 australia
-rw-r--r-- 1 root root 0 map 29 12:54 feathers
-rw-r--r-- 1 root root 0 map 29 12:54 my_os
drwxr-xr-x 1 root root 0 map 29 13:03 play
root@esdolghev:~/ex3# chmod g-x, g-r australia
chmod: неверный режим: «g-x,»
По команде «chmod --help» можно получить дополнительную информацию.
root@esdolghev:~/ex3# chmod g+rx australia
root@esdolghev:~/ex3# chmod o-x australia
root@esdolghev:~/ex3# ls -l
иторо 0
drwxr--r-- 1 root root 0 map 29 13:03 australia
-rw-r--r-- 1 root root 0 map 29 12:54 feathers
-rw-r--r-- 1 root root 0 map 29 12:54 my_os
drwxr-xr-x 1 root root 0 map 29 13:03 play
root@esdolghev:~/ex3# chmod g-r play
root@esdolghev:~/ex3# ls -l
иторо 0
drwxr--r-- 1 root root 0 map 29 13:03 australia
-rw-r--r-- 1 root root 0 map 29 12:54 feathers
-rw-r--r-- 1 root root 0 map 29 12:54 my_os
drwx--x--x 1 root root 0 map 29 13:03 play
root@esdolghev:~/ex3#

```

Рис. 3.7: Задание 3

```

drwxr--r-- 1 root root 0 map 29 13:03 australia
-rw-rw-r-- 1 root root 0 map 29 12:54 feathers
-r-xr--r-- 1 root root 0 map 29 12:54 my_os
drwx--x--x 1 root root 0 map 29 13:03 play
root@esdolghev:~/ex3#

```

Рис. 3.8: Задание 3

Задание 4 является небольшой самостоятельной работой. Вот результат её выполнения (рис. 3.9):

```

root@esdolgaev:~# cd ex3
root@esdolgaev:~/ex3# cp feathers file.old
root@esdolgaev:~/ex3# ls
australia feathers file.old my_os play
root@esdolgaev:~/ex3# mv flie.old play
mv: не удалось выполнить stat для 'flie.old': Нет такого файла или каталога
root@esdolgaev:~/ex3# mv file.old play
root@esdolgaev:~/ex3# cp play fun
cp: не указан -r; пропускается каталог 'play'
root@esdolgaev:~/ex3# cp -r play fun
root@esdolgaev:~/ex3# mv fun play/games
root@esdolgaev:~/ex3# ls
australia feathers my_os play
root@esdolgaev:~/ex3# chmod u-r feathers
root@esdolgaev:~/ex3# cat feathers
root@esdolgaev:~/ex3# cp feathers feathers1
root@esdolgaev:~/ex3# ls
australia feathers feathers1 my_os play
root@esdolgaev:~/ex3# chmod u+r feathers
root@esdolgaev:~/ex3# chmod u-x play
root@esdolgaev:~/ex3# cd play
root@esdolgaev:~/ex3/play# ls
file.old games
root@esdolgaev:~/ex3/play# cd ..
root@esdolgaev:~/ex3# chmod u+x play
root@esdolgaev:~/ex3#

```

Рис. 3.9: Задание 4

## 3.1 Краткая характеристика команд mount, fsck, mkfs и kill

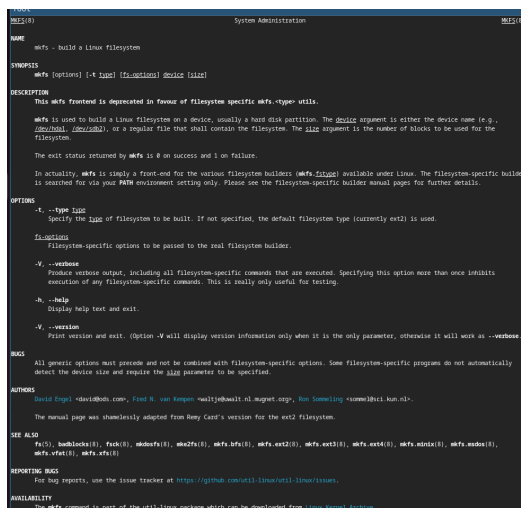
### 3.1.1 mount

Команда mount предназначена для подключения (монтирования) файловых систем и переносных накопителей (usb-флешка, карта памяти) к конкретным точкам монтирования в дереве директорий. При запуске без аргументов команда показывает все подключенные в данный момент файловые системы.



### 3.1.3 mkfs

Это универсальный инструмент, позволяющий форматировать диски и разделы с заданной файловой системой. Если вы устанавливаете новый диск, создаете новый раздел или переформатируете существующий, `mkfs` – незаменимая команда для настройки файловой системы.



```
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/sda1, /dev/cvtda), 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.-types) 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.

    -V, --version
        Print version and exit. (Option -V will display version information only when it is the only parameter, otherwise it will work as --verbose.)

BUGS
    All generic options must precede and not be combined with filesystem-specific options. Some filesystem-specific programs do not automatically detect the device size and require the size parameter to be specified.

AUTHORS
    David Engel <dsiel@bbs.com>, Fred N. van Kempen <waltje@u.waltje.nl>, Robert Siegel <robert@u.waltje.nl>.

    The manual page was shamelessly adapted from Remy Card's version for the ext2 filesystem.

SEE ALSO
    fs(2), badblocks(1), fsck(1), mdadmfs(1), mkcifs(1), mkfs.bfs(1), mkfs.ext2(1), mkfs.ext3(1), mkfs.ext4(1), mkfs.minix(1), mkfs.msdos(1), mkfs.vfat(1), mkfs.xfs(1).

REPORTING BUGS
    For bug reports, use the issue tracker at https://github.com/util-linux/util-linux/issues.

AVAILABILITY
    The mkfs command is part of the util-linux package which can be downloaded from http://linux.dedim.ch/linux/.
```

Рис. 3.12: `mkfs`

### 3.1.4 kill

Она позволяют прекратить выполнение запущенных процессов.

Команда `kill` отправляет сигнал (по умолчанию сигнал `SIGTERM`) запущенному процессу. Это действие по умолчанию обычно останавливает процессы. Если вы хотите остановить процесс, укажите идентификатор процесса (PID) в переменной `ProcessID`.

```

kill(1)                                User Commands                                kill(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal=- signal=-#] [-q value] [-s] [--timeout milliseconds signal] [--] pid/name ...
    kill -l [number] | -L

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

    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 to come 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:
    0
    where p is larger than 0. The process with PID p is signaled.
    *
    All processes in the current process group are signaled.
    -1
    All processes with a PID larger than 1 are signaled.
    -0
    where p is larger than 1. All processes in process group p are signaled. When an argument of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '-' option, otherwise it will be taken as the signal to send.

NAME
    All processes involved using this name will be signaled.

OPTIONS
    -s, --signal signal
    The signal to send. It may be given as a name or a number.
    -l, --list [number]
    Print a list of signal names, or convert the given signal number to a name. The signals can be found in man7.org/linux/signal.7.
    -L, --table
    Similar to -l, but it will print signal names and their corresponding numbers.

```

Рис. 3.13: kill

## 4 Ответы на контрольные вопросы

### 4.0.1 Вопрос 1

- 1) `sysfs` — виртуальная файловая система в операционной системе Linux. Экспортирует в пространство пользователя информацию ядра Linux о присутствующих в системе устройствах и драйверах. Впервые появилась в ядре версии 2.6. Необходимость создания была вызвана устаревшей системой работы ядра с устройствами.
- 2) `TraceFS` — новая файловая система для ядра Linux, ориентированная на подсистему трассировки

### 4.0.2 Вопрос 2

Файловая система в Linux определяет также организацию расположения файлов, по сути представляя собой иерархическую структуру «дерева»: начинается с корневого каталога «/» и разрастается ветвями в зависимости от работы системы.





`/lib` — содержит системные библиотеки, с которыми работают программы и модули ядра;

`/lost+found` — содержит файлы, восстановленные после сбоя работы системы. Система проведет проверку после сбоя и найденные файлы можно будет посмотреть в данном каталоге;

`/media` — точка монтирования внешних носителей. Например, когда вы вставляете диск в дисковод, он будет автоматически смонтирован в директорию `/media/cdrom`;

`/mnt` — точка временного монтирования. Файловые системы подключаемых устройств обычно монтируются в этот каталог для временного использования;

`/opt` — тут расположены дополнительные (необязательные) приложения. Такие программы обычно не подчиняются принятой иерархии и хранят свои файлы в одном подкаталоге (бинарные, библиотеки, конфигурации);

`/proc` — содержит файлы, хранящие информацию о запущенных процессах и о состоянии ядра ОС;

`/root` — директория, которая содержит файлы и личные настройки суперпользователя;

`/run` — содержит файлы состояния приложений. Например, PID-файлы или UNIX-сокеты;

`/sbin` — аналогично `/bin` содержит бинарные файлы. Утилиты нужны для настройки и администрирования системы суперпользователем;

`/srv` — содержит файлы сервисов, предоставляемых сервером (прим. FTP или Apache HTTP);

`/sys` — содержит данные непосредственно о системе. Тут можно узнать информацию о ядре, драйверах и устройствах;

`/tmp` — содержит временные файлы. Данные файлы доступны всем пользователям на чтение и запись. Стоит отметить, что данный каталог очи-

щается при перезагрузке;

`/usr` — содержит пользовательские приложения и утилиты второго уровня, используемые пользователями, а не системой. Содержимое доступно только для чтения (кроме `root`). Каталог имеет вторичную иерархию и похож на корневой;

`/var` — содержит переменные файлы. Имеет подкаталоги, отвечающие за отдельные переменные. Например, логи будут храниться в `/var/log`, кэш в `/var/cache`, очереди заданий в `/var/spool/` и так далее.

### 4.0.3 Вопрос 3

Монтирование тома.

### 4.0.4 Вопрос 4

Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

1. Один блок адресуется несколькими `inode` (принадлежит нескольким файлам).
2. Блок помечен как свободный, но в то же время занят (на него ссылается `inode`).
3. Блок помечен как занятый, но в то же время свободен (ни один `inode` на него не ссылается).
4. Неправильное число ссылок в `inode` (недостаток или избыток ссылающихся записей в каталогах).

5. Несовпадение между размером файла и суммарным размером адресуемых inode блоков.
6. Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы).
7. “Потерянные” файлы (правильные inode, на которые не ссылаются записи каталогов).
8. Недопустимые или неразмещенные номера inode в записях каталогов.

#### **4.0.5 Вопрос 5**

`mkfs` - позволяет создать файловую систему Linux.

#### **4.0.6 Вопрос 6**

`cat` - выводит содержимое файла на стандартное устройство вывода

#### **4.0.7 Вопрос 7**

`cp` – копирует или перемещает директорию, файлы.

#### **4.0.8 Вопрос 8**

`mv` - переименовать или переместить файл или директорию

#### **4.0.9 Вопрос 9**

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога)

или пользователь с правами администратора.

## **5 Выводы**

В ходе выполнения лабораторной работы я приобрел 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**