

Операционные системы

Анализ файловой структуры UNIX. Команды для работы с файлами и каталогами

Чернятин Артём Андреевич

2025-12-09

Содержание i

1. Цели и задачи работы
2. Процесс выполнения лабораторной работы
3. Выводы по проделанной работе

1. Цели и задачи работы

1.1 Цель лабораторной 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, по проверке использования диска и обслуживанию файловой системы.

1.2 Задачи лабораторной работы

- 1 Выполнить примеры
- 2 Выполнить действия по работе с каталогами и файлами
- 3 Выполнить действия с правами доступа
- 4 Получить дополнительные сведения при помощи справки по командам.

2. Процесс выполнения лабораторной работы

2.1 Выполнение примеров

```
aachernyatin@aachernyatin:~$  
aachernyatin@aachernyatin:~$ cd  
aachernyatin@aachernyatin:~$ touch abc1  
aachernyatin@aachernyatin:~$ cp abc1 april  
aachernyatin@aachernyatin:~$ cp abc1 may  
aachernyatin@aachernyatin:~$ mkdir monthly  
aachernyatin@aachernyatin:~$ cp april may monthly/  
aachernyatin@aachernyatin:~$ cp monthly/may monthly/june  
aachernyatin@aachernyatin:~$ ls monthly/  
april june may  
aachernyatin@aachernyatin:~$ mkdir monthly.00  
aachernyatin@aachernyatin:~$ cp -r monthly monthly.00/  
aachernyatin@aachernyatin:~$ cp -r monthly.00/ /tmp  
aachernyatin@aachernyatin:~$ █
```

Рисунок 1: Выполнение примеров

2.2 Выполнение примеров

```
aachernyatin@aachernyatin:~$ cd  
aachernyatin@aachernyatin:~$ mv april july  
aachernyatin@aachernyatin:~$ mv july monthly.00/  
aachernyatin@aachernyatin:~$ ls monthly.00/  
july monthly  
aachernyatin@aachernyatin:~$ mv monthly.00/ monthly.01  
aachernyatin@aachernyatin:~$ mkdir reports  
aachernyatin@aachernyatin:~$ mv monthly.01/ reports/  
aachernyatin@aachernyatin:~$ mv reports/monthly.01/ reports/monthly  
aachernyatin@aachernyatin:~$ █
```

Рисунок 2: Выполнение примеров

2.3 Выполнение примеров

```
aachernyatin@aachernyatin:~$  
aachernyatin@aachernyatin:~$ touch may  
aachernyatin@aachernyatin:~$ ls -l may  
-rw-r--r--. 1 aachernyatin aachernyatin 0 Dec  9 15:37 may  
aachernyatin@aachernyatin:~$ chmod u+x may  
aachernyatin@aachernyatin:~$ ls -l may  
-rwxr--r--. 1 aachernyatin aachernyatin 0 Dec  9 15:37 may  
aachernyatin@aachernyatin:~$ chmod u-x may  
aachernyatin@aachernyatin:~$ ls -l may  
-rw-r--r--. 1 aachernyatin aachernyatin 0 Dec  9 15:37 may  
aachernyatin@aachernyatin:~$ cd  
aachernyatin@aachernyatin:~$ chmod g-r,o-r monthly/  
aachernyatin@aachernyatin:~$ chmod g+w abc1  
aachernyatin@aachernyatin:~$ █
```

Рисунок 3: Выполнение примеров

2.4 Создание директорий и копирование файлов

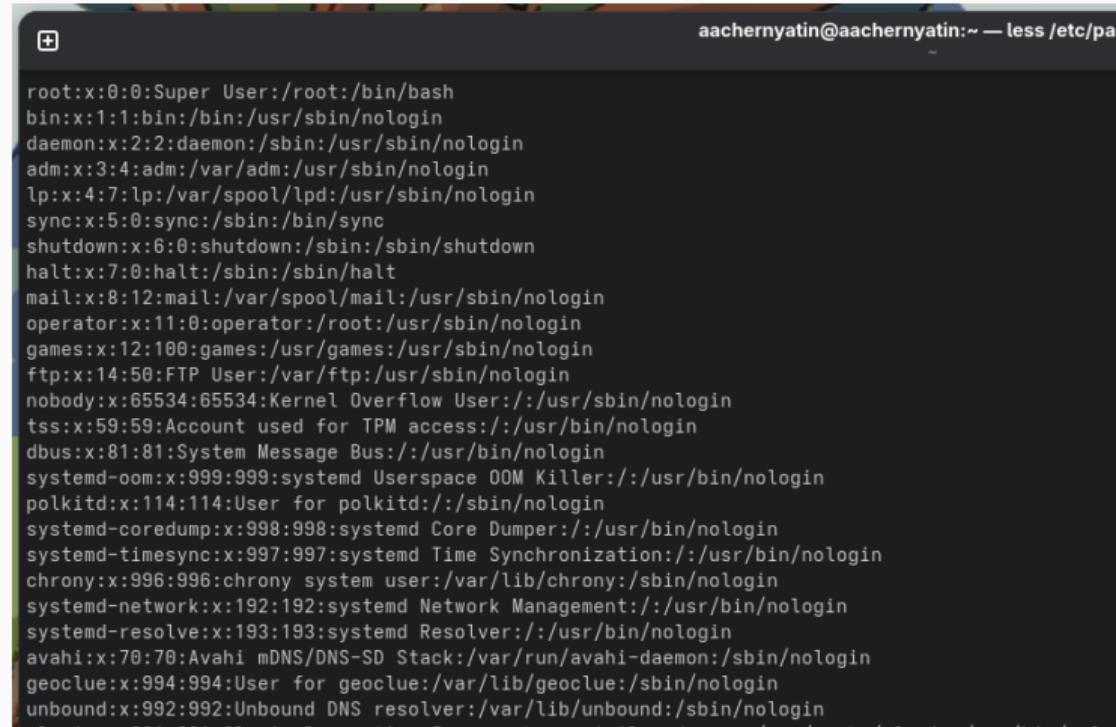
```
aachernyatin@aachernyatin:~$ cp /usr/include/linux/sysinfo.h ~
aachernyatin@aachernyatin:~$ mv sysinfo.h equipment
aachernyatin@aachernyatin:~$ mkdir ski.plases
aachernyatin@aachernyatin:~$ mv equipment ski.plases/
aachernyatin@aachernyatin:~$ mv ski.plases/equipment ski.plases/quiplist
aachernyatin@aachernyatin:~$ touch abc1
aachernyatin@aachernyatin:~$ cp abc1 Ski.plases/quiplist2
aachernyatin@aachernyatin:~$ cd ski.plases/
aachernyatin@aachernyatin:~/ski.plases$ mkdir equipment
aachernyatin@aachernyatin:~/ski.plases$ ls
equipment quiplist quiplist2
aachernyatin@aachernyatin:~/ski.plases$ mv quiplist equipment/
aachernyatin@aachernyatin:~/ski.plases$ mv quiplist2 equipment/
aachernyatin@aachernyatin:~/ski.plases$ cd
aachernyatin@aachernyatin:~$ mkdie newdir
bash: mkdie: command not found...
aachernyatin@aachernyatin:~$ mkdir newdir
aachernyatin@aachernyatin:~$ mv newdir ski.plases/
aachernyatin@aachernyatin:~$ mv ski.plases/newdir/ ski.plases/plans
aachernyatin@aachernyatin:~$ █
```

Рисунок 4: Работа с каталогами

2.5 Работа с командой chmod

```
aachernyatin@aachernyatin:~$  
aachernyatin@aachernyatin:~$ mkdir australia play  
aachernyatin@aachernyatin:~$ touch my_os feathers  
aachernyatin@aachernyatin:~$ chmod 744 australia/  
aachernyatin@aachernyatin:~$ chmod 711 play/  
aachernyatin@aachernyatin:~$ chmod 544 my_os  
aachernyatin@aachernyatin:~$ chmod 664 feathers  
aachernyatin@aachernyatin:~$ ls -l  
total 0  
-rw-rw-r--. 1 aachernyatin aachernyatin 0 Dec 9 15:41 abc1  
drwxr--r--. 1 aachernyatin aachernyatin 0 Dec 9 15:43 australia  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Desktop  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Documents  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Downloads  
-rw-rw-r--. 1 aachernyatin aachernyatin 0 Dec 9 15:44 feathers  
drwxr-xr-x. 1 aachernyatin aachernyatin 74 Dec 8 17:55 git-extended  
-rw-r--r--. 1 aachernyatin aachernyatin 0 Dec 9 15:37 may  
drwx--x--x. 1 aachernyatin aachernyatin 24 Dec 9 15:32 monthly  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Music  
-r-xr--r--. 1 aachernyatin aachernyatin 0 Dec 9 15:44 my_os  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Pictures  
drwx--x--x. 1 aachernyatin aachernyatin 0 Dec 9 15:43 play  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Public  
drwxr-xr-x. 1 aachernyatin aachernyatin 14 Dec 9 15:35 reports  
drwxr-xr-x. 1 aachernyatin aachernyatin 28 Dec 9 15:42 ski.plases  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Templates  
drwxr-xr-x. 1 aachernyatin aachernyatin 0 Dec 8 16:38 Videos  
drwxr-xr-x. 1 aachernyatin aachernyatin 10 Dec 8 17:08 work
```

2.6 Файл /etc/passwd



The screenshot shows a terminal window with the command `less /etc/passwd` running. The window title is "aachernyatin@aachernyatin:~ — less /etc/pas". The terminal displays a list of user entries from the `/etc/passwd` file. Each entry consists of five fields separated by colons: login name, password, user ID, group ID, and home directory and shell. The entries include root, bin, daemon, adm, lp, sync, shutdown, halt, mail, operator, games, ftp, nobody, tss, dbus, systemd-oom, polkitd, systemd-coredump, systemd-timesync, chrony, systemd-network, systemd-resolve, avahi, geoclue, and unbound.

```
aachernyatin@aachernyatin:~ — less /etc/pas
root:x:0:0:Super User:/bin/bash
bin:x:1:1:bin:/bin:/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
tss:x:59:59:Account used for TPM access:/:/usr/bin/nologin
dbus:x:81:81:System Message Bus:/:/usr/bin/nologin
systemd-oom:x:999:999:systemd Userspace OOM Killer:/:/usr/bin/nologin
polkitd:x:114:114:User for polkitd:/:/sbin/nologin
systemd-coredump:x:998:998:systemd Core Dumper:/:/usr/bin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/bin/nologin
chrony:x:996:996:chrony system user:/var/lib/chrony:/sbin/nologin
systemd-network:x:192:192:systemd Network Management:/:/usr/bin/nologin
systemd-resolve:x:193:193:systemd Resolver:/:/usr/bin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
geoclue:x:994:994:User for geoclue:/var/lib/geoclue:/sbin/nologin
unbound:x:992:992:Unbound DNS resolver:/var/lib/unbound:/sbin/nologin
```

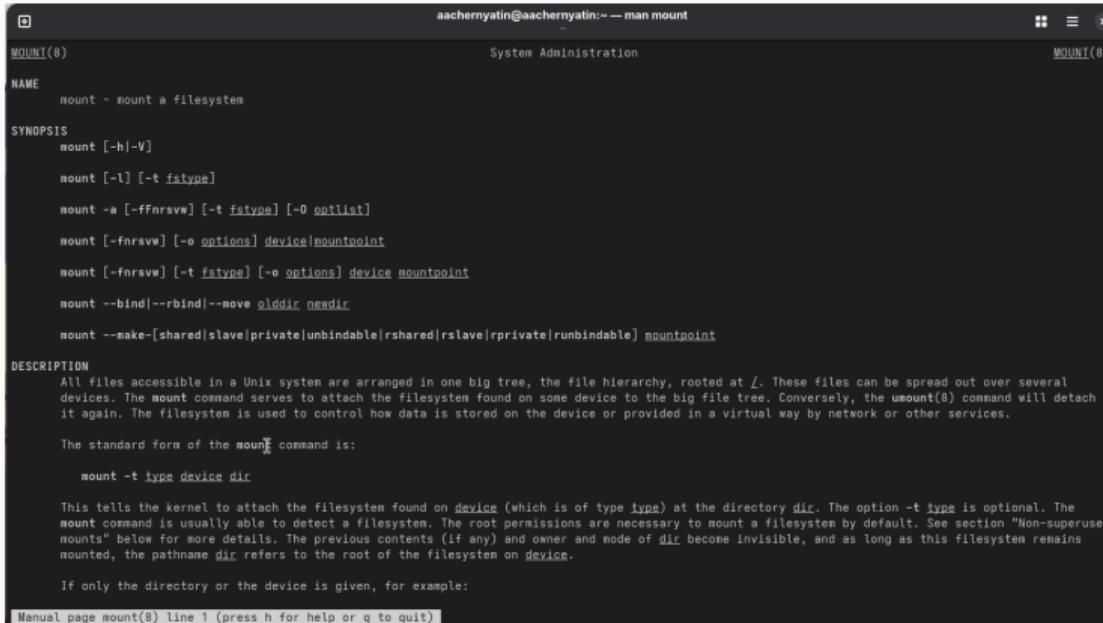
Рисунок 6: Файл /etc/passwd

2.7 Работа с файлами и правами доступа

```
aachernyatin@aachernyatin:~$ cp feathers file.old
aachernyatin@aachernyatin:~$ mv file.old play/
aachernyatin@aachernyatin:~$ mkdir fun
aachernyatin@aachernyatin:~$ cp -R play/ fun/
aachernyatin@aachernyatin:~$ mv fun/ play/games
aachernyatin@aachernyatin:~$ chmod u-r feathers
aachernyatin@aachernyatin:~$ cat feathers
cat: feathers: Permission denied
aachernyatin@aachernyatin:~$ chmod u+r feathers
aachernyatin@aachernyatin:~$ chmod u-x play/
aachernyatin@aachernyatin:~$ cd play/
bash: cd: play/: Permission denied
aachernyatin@aachernyatin:~$ chmod +x play/
aachernyatin@aachernyatin:~$ █
```

Рисунок 7: Работа с файлами и правами доступа

2.8 Справка по командам



The screenshot shows a terminal window with the title bar "aichernyatin@aichernyatin: ~ man mount". The window displays the man page for the "mount" command. The page is titled "MOUNT(8)" and is part of the "System Administration" section. It includes sections for "NAME", "SYNOPSIS", "DESCRIPTION", and examples. The "SYNOPSIS" section lists various command-line options and arguments. The "DESCRIPTION" section explains the purpose of the command and how it integrates into the Unix file hierarchy. Examples show how to mount a filesystem from a device or network location.

```
aichernyatin@aichernyatin: ~ 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 needir
       mount --make-[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable] mountpoint

DESCRIPTION
       All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /. These files can be spread out over several devices. The mount command serves to attach 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.

       If only the directory or the device is given, for example:
       mount device dir

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

Рисунок 8: Команда mount

2.9 Справка по командам

```
aichernyatin@aichernyatin:~ — man fsck
FSCK(8)                                     System Administration                               FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [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.

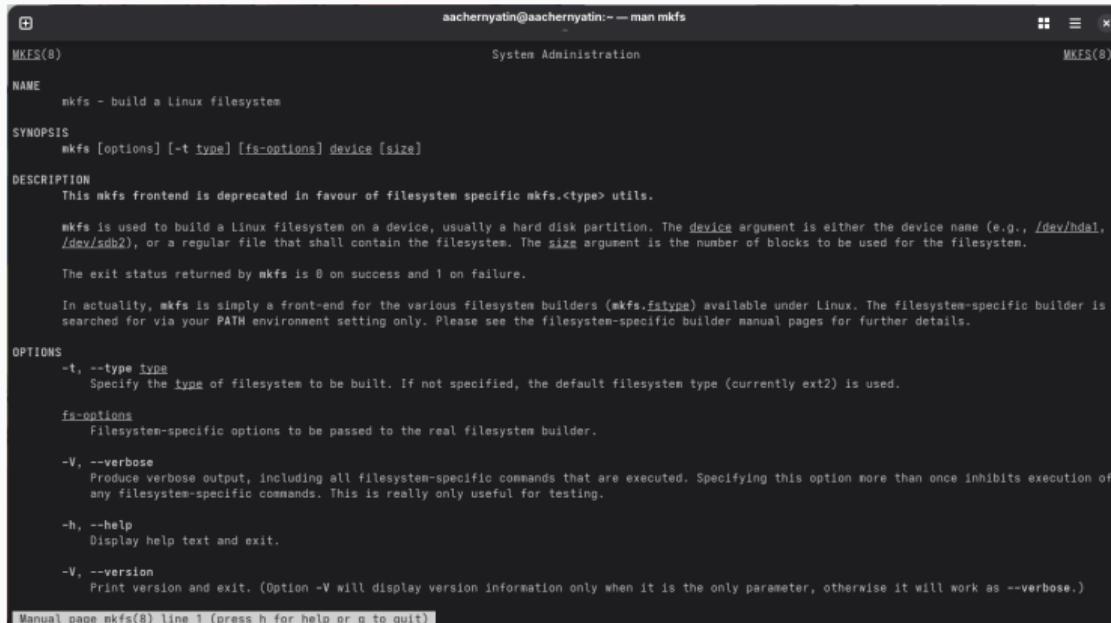
    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to checking filesystems in /etc/fstab serially. This is equivalent to the -As options.

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

    0          No errors
    1          Filesystem errors corrected
    2          System should be rebooted
    4          Filesystem errors left uncorrected
    8          Operational error
    16         Usage or syntax error
Manual page fsck(8) line 1 (press h for help or q to quit)
```

Рисунок 9: Команда fsck

2.10 Справка по командам



```
aichernyatin@aichernyatin:~ -- 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.

    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.

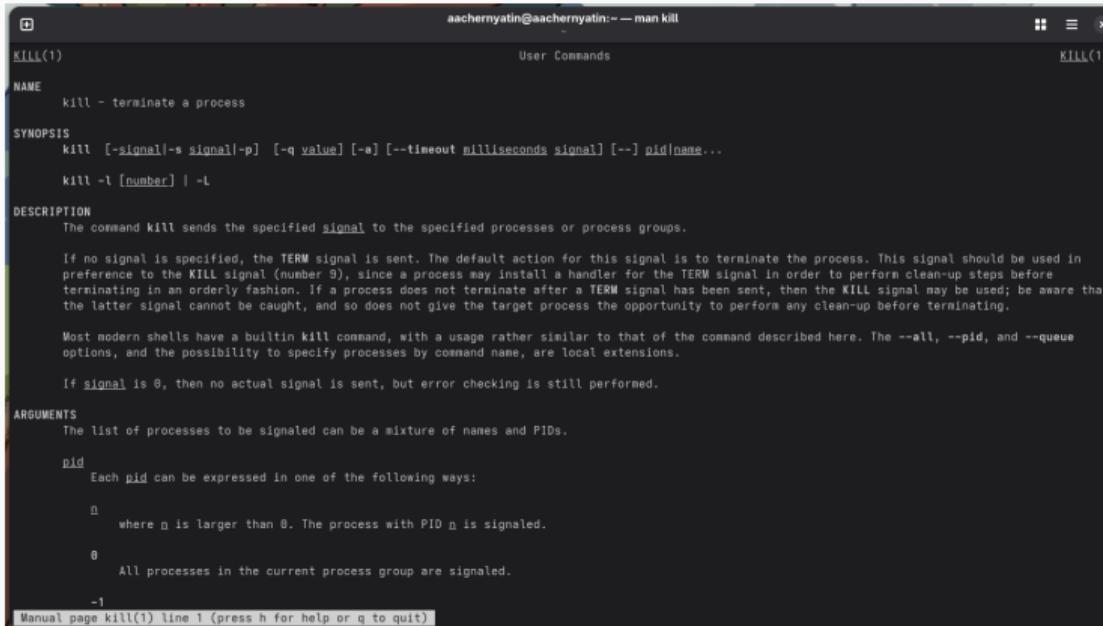
    -h, --help
        Display help text and exit.

    -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.)

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

Рисунок 10: Команда mkfs

2.11 Справка по командам



The screenshot shows a terminal window with the title "aachernyatin@aachernyatin: ~ man kill". The window displays the man page for the kill command. The page is titled "KILL(1)" and includes sections for NAME, SYNOPSIS, DESCRIPTION, and ARGUMENTS. The SYNOPSIS section shows the command syntax: "kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--] pid|name...". The DESCRIPTION section explains that the command sends a specified signal to processes. It notes that if no signal is specified, TERM is sent. It also mentions that if a process does not terminate after a TERM signal, KILL can be used. The ARGUMENTS section details how to specify processes using PIDs or names, and it defines the options -a, -q, --timeout, and --.

```
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:
            0           where 0 is larger than 0. The process with PID 0 is signaled.
            @           All processes in the current process group are signaled.
            -1          Manual page kill(1) line 3 (press h for help or q to quit)
```

Рисунок 11: Команда kill

3. Выводы по проделанной работе

3.1 Вывод

В ходе данной работы мы ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. Научились совершать базовые операции с файлами, управлять правами их доступа для пользователя и групп. Ознакомились с Анализом файловой системы. А также получили базовые навыки по проверке использования диска и обслуживанию файловой системы.