

# Операционные системы

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

---

Хатамов Эзиз

28 марта 2025

Российский университет дружбы народов, Москва, Россия

## Цели и задачи работы

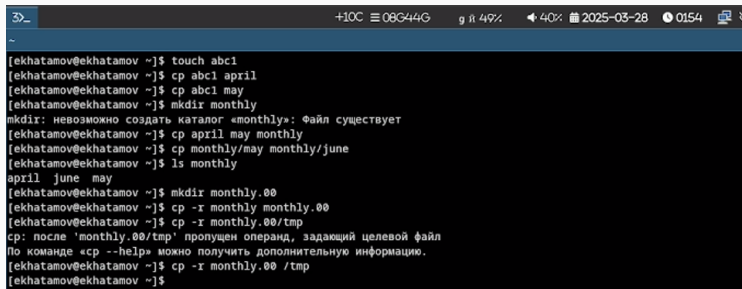
---

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

## Процесс выполнения лабораторной работы

---



```
[ekhatamov@ekhatamov ~]$ touch abc1
[ekhatamov@ekhatamov ~]$ cp abc1 april
[ekhatamov@ekhatamov ~]$ cp abc1 may
[ekhatamov@ekhatamov ~]$ mkdir monthly
mkdir: невозможно создать каталог «monthly»: Файл существует
[ekhatamov@ekhatamov ~]$ cp april may monthly
[ekhatamov@ekhatamov ~]$ cp monthly/may monthly/june
[ekhatamov@ekhatamov ~]$ ls monthly
april  june  may
[ekhatamov@ekhatamov ~]$ mkdir monthly.00
[ekhatamov@ekhatamov ~]$ cp -r monthly monthly.00
[ekhatamov@ekhatamov ~]$ cp -r monthly.00/tmp
cp: после 'monthly.00/tmp' пропущен операнд, задающий целевой файл
По команде «cp --help» можно получить дополнительную информацию.
[ekhatamov@ekhatamov ~]$ cp -r monthly.00 /tmp
[ekhatamov@ekhatamov ~]$
```

Рис. 1: Выполнение примеров

```
[ekhatamov@ekhatamov ~]$ mv april july
[ekhatamov@ekhatamov ~]$ mv july monthly.00
[ekhatamov@ekhatamov ~]$ ls monthly.00
july  monthly
[ekhatamov@ekhatamov ~]$ mv monthly.00 monthly.01
[ekhatamov@ekhatamov ~]$ mkdir reports
[ekhatamov@ekhatamov ~]$ mv monthly.01 reports
[ekhatamov@ekhatamov ~]$ mv reports/monthly.01 reports/monthly
[ekhatamov@ekhatamov ~]$
```

Рис. 2: Выполнение примеров

```
[ekhatamov@ekhatamov ~]$ touch may
[ekhatamov@ekhatamov ~]$ ls -l may
-rw-r--r--. 1 ekhatamov ekhatamov 0 map 28 01:56 may
[ekhatamov@ekhatamov ~]$ chmod u+x may
[ekhatamov@ekhatamov ~]$ ls -l may
-rwxr--r--. 1 ekhatamov ekhatamov 0 map 28 01:56 may
[ekhatamov@ekhatamov ~]$ chmod u-x may
[ekhatamov@ekhatamov ~]$ ls -l may
-rw-r--r--. 1 ekhatamov ekhatamov 0 map 28 01:56 may
[ekhatamov@ekhatamov ~]$ chmod g-r,o-r monthly
[ekhatamov@ekhatamov ~]$ chmod g+w abc1
[ekhatamov@ekhatamov ~]$
```

Рис. 3: Выполнение примеров



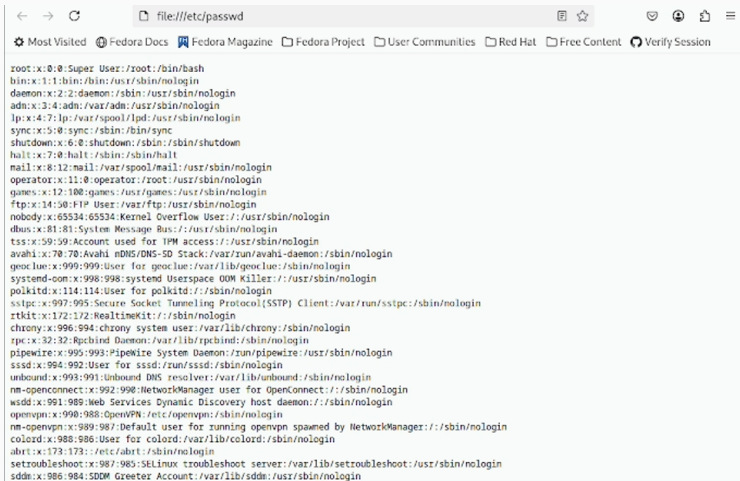
## Создание директорий и копирование файлов

```
[ekhatamov@ekhatamov ~]$ cp /usr/include/linux/sysinfo.h ~
[ekhatamov@ekhatamov ~]$ mv sysinfo.h equipment
[ekhatamov@ekhatamov ~]$ mkdir ski.plases
[ekhatamov@ekhatamov ~]$ mv equipment ski.plases/
[ekhatamov@ekhatamov ~]$ mv ski.plases/equipment ski.plases/equiplist
mv: не удалось выполнить stat для 'ski.plases/equipment': Нет такого файла или каталога
[ekhatamov@ekhatamov ~]$ mv ski.plases/equipment ski.plases/equiplist
[ekhatamov@ekhatamov ~]$ touch abc1
[ekhatamov@ekhatamov ~]$ cp abc1 ski.plases/equiplist2
[ekhatamov@ekhatamov ~]$ cd ski.plases/
[ekhatamov@ekhatamov ski.plases]$ mkdir equiplist equipment/
mkdir: невозможно создать каталог «equiplist»: Файл существует
[ekhatamov@ekhatamov ski.plases]$ mkdir equiplist2 equipment/
mkdir: невозможно создать каталог «equiplist2»: Файл существует
mkdir: невозможно создать каталог «equipment/»: Файл существует
[ekhatamov@ekhatamov ski.plases]$ mkdir newdir
[ekhatamov@ekhatamov ski.plases]$ mkdir equipment
mkdir: невозможно создать каталог «equipment»: Файл существует
[ekhatamov@ekhatamov ski.plases]$ mkdir equiplist equipment/
mkdir: невозможно создать каталог «equiplist»: Файл существует
mkdir: невозможно создать каталог «equipment/»: Файл существует
[ekhatamov@ekhatamov ski.plases]$ mkdir equiplist2 equipment/
mkdir: невозможно создать каталог «equiplist2»: Файл существует
mkdir: невозможно создать каталог «equipment/»: Файл существует
[ekhatamov@ekhatamov ski.plases]$ mv newdir ski.plases/newdir/ ski.plases/plans
mv: cannot mkdir 'ski.plases/plans': Нет такого файла или каталога
```

Рис. 4: Работа с каталогами

```
[ekhatamov@ekhatamov ski.plases]$  
[ekhatamov@ekhatamov ski.plases]$ mkdir australia play  
[ekhatamov@ekhatamov ski.plases]$ touch my_os feathers  
[ekhatamov@ekhatamov ski.plases]$ chmod 744 australia/  
[ekhatamov@ekhatamov ski.plases]$ chmod 711 play/  
[ekhatamov@ekhatamov ski.plases]$ chmod 544 my_os  
[ekhatamov@ekhatamov ski.plases]$ touch 644 feathers  
[ekhatamov@ekhatamov ski.plases]$ ls -l  
итого 4  
-rw-r--r--. 1 ekhatamov ekhatamov      0 map 28 02:04 644  
drwxr--r--. 1 ekhatamov ekhatamov      0 map 28 02:03 australia  
-rw-r--r--. 1 ekhatamov ekhatamov 1049 map 28 01:58 equiplist  
-rw-r--r--. 1 ekhatamov ekhatamov      0 map 28 02:00 equiplist2  
drwxr-xr-x. 1 ekhatamov ekhatamov      0 map 28 02:00 equipment  
-rw-r--r--. 1 ekhatamov ekhatamov      0 map 28 02:04 feathers  
-r-xr--r--. 1 ekhatamov ekhatamov      0 map 28 02:04 my_os  
drwxr-xr-x. 1 ekhatamov ekhatamov      0 map 28 02:01 newdir  
drwx--x--x. 1 ekhatamov ekhatamov      0 map 28 02:03 play  
[ekhatamov@ekhatamov ski.plases]$
```

Рис. 5: Настройка прав доступа



```
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
ssltcp:x:997:995:Secure Socket Tunneling Protocol (SSTP) Client:/var/run/ssltcp:/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
wsdd: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
abrt:x:173:173:/etc/abrt:/sbin/nologin
setroubleshoot:x:987:985:SELinux troubleshoot server:/var/lib/setroubleshoot:/usr/sbin/nologin
sddm:x:986:984:SDM Greeter Account:/var/lib/sddm:/usr/sbin/nologin
```

Рис. 6: Файл /etc/passwd

```
[ekhatamov@ekhatamov ski.plases]$ cp feathers file.old
[ekhatamov@ekhatamov ski.plases]$ mv file.old
mv: после 'file.old' пропущен операнд, задающий целевой файл
По команде «mv --help» можно получить дополнительную информацию.
[ekhatamov@ekhatamov ski.plases]$ mv file.old play/
[ekhatamov@ekhatamov ski.plases]$ mkdir fun
[ekhatamov@ekhatamov ski.plases]$ cp -R play/ fun/
[ekhatamov@ekhatamov ski.plases]$ mv fun/ play/games
[ekhatamov@ekhatamov ski.plases]$ chmod u-r feathers
[ekhatamov@ekhatamov ski.plases]$ cat feathers
cat: feathers: Отказано в доступе
[ekhatamov@ekhatamov ski.plases]$ cp feathers feathers2
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[ekhatamov@ekhatamov ski.plases]$ chmod u+r feathers
[ekhatamov@ekhatamov ski.plases]$ chmod u-r play/
[ekhatamov@ekhatamov ski.plases]$ cd play/
[ekhatamov@ekhatamov play]$ chmod *x play/
chmod: неверный режим: «*x»
По команде «chmod --help» можно получить дополнительную информацию.
[ekhatamov@ekhatamov play]$ chmod +x play/
chmod: невозможно получить доступ к 'play/': Нет такого файла или каталога
[ekhatamov@ekhatamov play]$ cd ski.plases
bash: cd: ski.plases: Нет такого файла или каталога
[ekhatamov@ekhatamov play]$ chmod +x play/
chmod: невозможно получить доступ к 'play/': Нет такого файла или каталога
[ekhatamov@ekhatamov play]$ cd play/
```

```
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|xprivate|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

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

```
3>_ +10C 08G44G g 65% 195% 2025-03-28 0217
man fsck
FSCK(8) System Administration FSCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-x [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

    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)
```

```
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.

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

Рис. 10: Команда mkfs

```
3>_ +10C 08G44G g 66% 195% 2025-03-28 0220
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.
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.
Manual page kill(1) line 1 (press h for help or q to quit)
```



## Выводы по проделанной работе

---

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.