

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

Умирзоков Иброхим Шомилжон угли¹

22 июня, 2024, Москва, Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

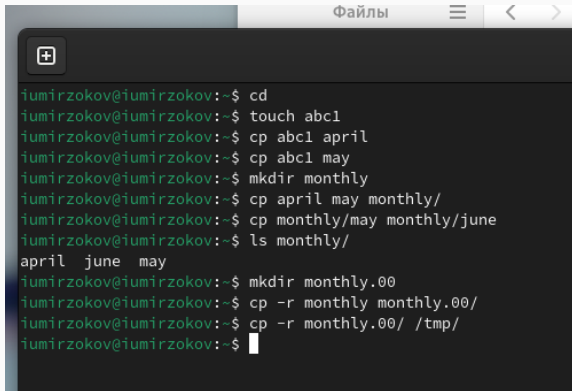
Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами, по проверке использования диска и обслуживанию файловой системы.

Задачи лабораторной работы

- 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

Выполнение примеров



The image shows a terminal window with a title bar containing the word "Файлы" (Files) and navigation icons. The terminal has a dark background with green text. It displays a sequence of commands entered by a user named iumirzokov, along with the output of some of these commands. The commands include creating a file, copying it to different locations, creating a directory, and listing the contents of a directory.

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

Рис. 1: Выполнение примеров

Выполнение примеров

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

Рис. 2: Выполнение примеров

Выполнение примеров

```
iumirzokov@iumirzokov:~$  
iumirzokov@iumirzokov:~$ cd  
iumirzokov@iumirzokov:~$ touch may  
iumirzokov@iumirzokov:~$ ls -l may  
-rw-r--r--. 1 iumirzokov iumirzokov 0 июн 22 10:50 may  
iumirzokov@iumirzokov:~$ chmod u+x may  
iumirzokov@iumirzokov:~$ ls -l may  
-rwxr--r--. 1 iumirzokov iumirzokov 0 июн 22 10:50 may  
iumirzokov@iumirzokov:~$ chmod u-x may  
iumirzokov@iumirzokov:~$ ls -l may  
-rw-r--r--. 1 iumirzokov iumirzokov 0 июн 22 10:50 may  
iumirzokov@iumirzokov:~$ cd  
iumirzokov@iumirzokov:~$ chmod g-r,o-r monthly/  
iumirzokov@iumirzokov:~$ chmod g+w abc1  
iumirzokov@iumirzokov:~$
```

Рис. 3: Выполнение примеров

Создание директорий и копирование файлов

```
iumirzokov@iumirzokov:~$  
iumirzokov@iumirzokov:~$ cp /usr/include/linux/sysinfo.h ~  
iumirzokov@iumirzokov:~$ mv sysinfo.h equipment  
iumirzokov@iumirzokov:~$ mkdir ski.plases  
iumirzokov@iumirzokov:~$ mv equipment ski.plases/  
iumirzokov@iumirzokov:~$ mv ski.plases/equipment ski.plases/equiplist  
iumirzokov@iumirzokov:~$ touch abc1  
iumirzokov@iumirzokov:~$ cp abc1 ski.plases/equiplist2  
iumirzokov@iumirzokov:~$ cd ski.plases/  
iumirzokov@iumirzokov:~/ski.plases$ mkdir equipment  
iumirzokov@iumirzokov:~/ski.plases$ mv equiplist equipment/  
iumirzokov@iumirzokov:~/ski.plases$ mv equiplist2 equipment/  
iumirzokov@iumirzokov:~/ski.plases$ cd  
iumirzokov@iumirzokov:~$ mkdir newdir  
iumirzokov@iumirzokov:~$ mv newdir/ ski.plases/  
iumirzokov@iumirzokov:~$ mv ski.plases/newdir/ ski.plases/plans  
iumirzokov@iumirzokov:~$
```

Рис. 4: Работа с каталогами

Работа с командой chmod

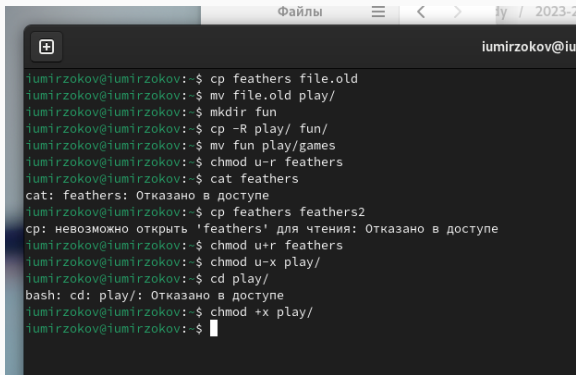
```
iumirzokov@iumirzokov:~$  
iumirzokov@iumirzokov:~$ mkdir australia play  
iumirzokov@iumirzokov:~$ touch my_os feathers  
iumirzokov@iumirzokov:~$ chmod 744 australia/  
iumirzokov@iumirzokov:~$ chmod 711 play  
iumirzokov@iumirzokov:~$ chmod 544 my_os  
iumirzokov@iumirzokov:~$ chmod 664 feathers  
iumirzokov@iumirzokov:~$ ls -l  
итого 0  
-rw-rw-r--. 1 iumirzokov iumirzokov 0 июн 22 10:52 abc1  
drwxr--r--. 1 iumirzokov iumirzokov 0 июн 22 10:53 australia  
-rw-rw-r--. 1 iumirzokov iumirzokov 0 июн 22 10:53 feathers  
drwxr-xr-x. 1 iumirzokov iumirzokov 74 июн 21 21:55 git-extended  
-rw-r--r--. 1 iumirzokov iumirzokov 0 июн 22 10:50 may  
drwx--x--x. 1 iumirzokov iumirzokov 24 июн 22 10:45 monthly  
-r-xr--r--. 1 iumirzokov iumirzokov 0 июн 22 10:53 my_os  
drwx--x--x. 1 iumirzokov iumirzokov 0 июн 22 10:53 play  
drwxr-xr-x. 1 iumirzokov iumirzokov 14 июн 22 10:48 reports  
drwxr-xr-x. 1 iumirzokov iumirzokov 28 июн 22 10:53 ski.places  
drwxr-xr-x. 1 iumirzokov iumirzokov 10 июн 21 21:20 work  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Видео  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Документы  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Загрузки  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Изображения  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Музыка  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Общедоступные  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 'Рабочий стол'  
drwxr-xr-x. 1 iumirzokov iumirzokov 0 июн 21 21:08 Шаблоны  
iumirzokov@iumirzokov:~$
```

Рис. 5: Настройка прав доступа

Файл /etc/passwd

```
итого 1484
drwxr-xr-x. 1 root root 4750 июн 21 21:06 ./
dr-xr-xr-x. 1 root root 158 мар 8 19:41 ../
drwxr-xr-x. 1 root root 126 ноя 1 2023 abrt/
-rw-r--r--. 1 root root 16 фев 8 15:51 adjtime
-rw-r--r--. 1 root root 1529 июл 25 2023 aliases
drwxr-xr-x. 1 root root 70 янв 29 03:00 alsa/
drwxr-xr-x. 1 root root 1584 апр 5 10:26 alternatives/
drwxr-xr-x. 1 root root 56 ноя 1 2023 anaconda/
-rw-r--r--. 1 root root 269 июл 19 2023 anthy-unicode.conf
-rw-r--r--. 1 root root 833 фев 10 2023 appstream.conf
-rw-r--r--. 1 root root 55 янв 29 03:00 asound.conf
drwxr-xr-x. 1 root root 108 янв 24 03:00 audit/
drwxr-xr-x. 1 root root 232 фев 8 15:51 authselect/
drwxr-xr-x. 1 root root 66 ноя 1 2023 avahi/
drwxr-xr-x. 1 root root 84 ноя 1 2023 bash_completion.d/
-rw-r--r--. 1 root root 2638 июл 25 2023 bashrc
-rw-r--r--. 1 root root 535 янв 5 03:00 bindresvport.blacklist
drwxr-xr-x. 1 root root 0 мар 1 03:00 binfmt.d/
dr-xr-xr-x. 1 root root 62 мар 8 19:44 bluetooth/
-rw-r-----. 1 root brlapi 33 ноя 1 2023 brlapi.key
drwxr-xr-x. 1 root root 76 дек 20 2023 brlty/
-rw-r--r--. 1 root root 30583 дек 20 2023 brlty.conf
drwxr-xr-x. 1 root root 0 янв 10 03:00 ceph/
drwxr-xr-x. 1 root root 0 янв 17 03:00 chkconfig.d/
drwxr-xr-x. 1 root root 44 ноя 1 2023 chromium/
-rw-r--r--. 1 root root 1372 дек 5 2023 chrony.conf
drwxr-xr-x. 1 root root 24 янв 12 03:00 cifs-utils/
drwxr-xr-x. 1 root root 168 янв 29 03:00 containers/
drwx-----. 1 root root 0 мар 1 03:00 credstore/
drwx-----. 1 root root 0 мар 1 03:00 credstore.encrypted/
drwxr-xr-x. 1 root root 70 дек 4 2023 crypto-policies/
-rw-----. 1 root root 0 фев 8 15:49 crypttab
-rw-r--r--. 1 root root 1401 июл 25 2023 csh.cshrc
-rw-r--r--. 1 root root 1544 июл 25 2023 csh.login
drwxr-xr-x. 1 root lp 396 июн 22 10:23 cups/
/etc
```

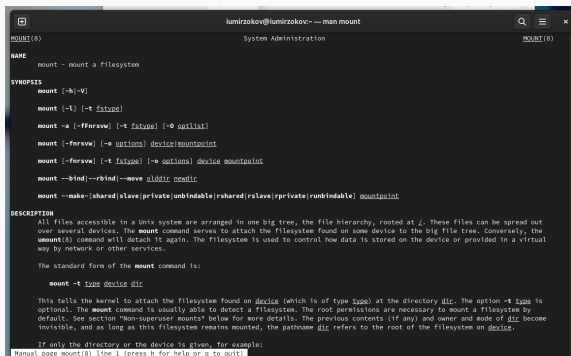
Работа с файлами и правами доступа



The screenshot shows a terminal window with a dark background. The title bar at the top contains the word 'Файлы' (Files), a hamburger menu icon, and navigation arrows. The address bar shows the path 'fy / 2023-2'. The terminal prompt is 'iumirzokov@iu'. The user enters a series of commands to create and manage files and directories, including copying, moving, creating, and setting permissions. Some commands result in error messages indicating permission issues.

```
iumirzokov@iumirzokov:~$ cp feathers file.old
iumirzokov@iumirzokov:~$ mv file.old play/
iumirzokov@iumirzokov:~$ mkdir fun
iumirzokov@iumirzokov:~$ cp -R play/ fun/
iumirzokov@iumirzokov:~$ mv fun play/games
iumirzokov@iumirzokov:~$ chmod u-r feathers
iumirzokov@iumirzokov:~$ cat feathers
cat: feathers: Отказано в доступе
iumirzokov@iumirzokov:~$ cp feathers feathers2
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
iumirzokov@iumirzokov:~$ chmod u+r feathers
iumirzokov@iumirzokov:~$ chmod u-x play/
iumirzokov@iumirzokov:~$ cd play/
bash: cd: play/: Отказано в доступе
iumirzokov@iumirzokov:~$ chmod +x play/
iumirzokov@iumirzokov:~$
```

Рис. 7: Работа с файлами и правами доступа



```

Iumirzokov@Iumirzokov: ~ — man mount
System Administration                                mount(8)

NAME
  mount - mount a filesystem

SYNOPSIS
  mount [-bj-V]

  mount [-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

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

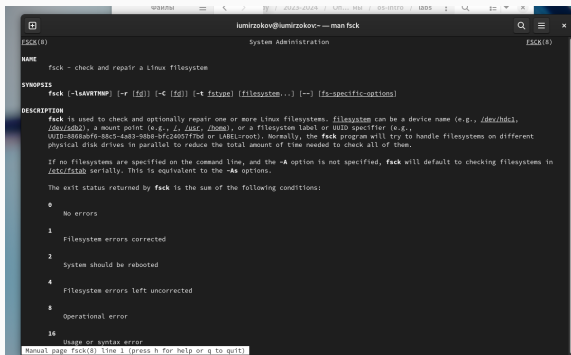
  This tells the kernel to attach the filesystem found on device (which is of type type) at the directory dir. The option -t type is
  optional. The mount command is usually able to detect a filesystem. The root permissions are necessary to mount a filesystem by
  default. See section "Non-superuser mounts" 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:

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

```

Рис. 8: Команда mount



```
lumirzokov@lumirzokov:~$ man fsck

fsck(8)                                System Administration                                fsck(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVTnmp] [-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=8808abf6-88c5-4a83-98b8-bfc240577fb6 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.

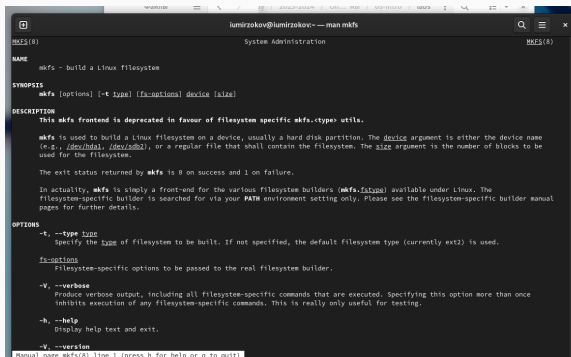
    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to checking filesystems in
    /etc/fstab serially. This is equivalent to the -As options.

    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)
```

Рис. 9: Команда fsck



```
lumirzokov@lumirzokov:~$ 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.ctype 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.filesystem) 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.

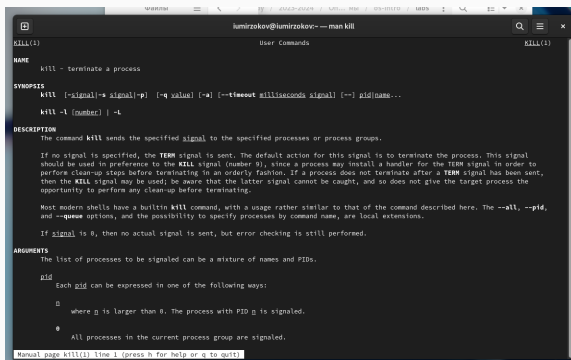
    -V, --verbose
        Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.

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

    -V, --version
        Display the version number and exit.

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

Рис. 10: Команда mkfs



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

        0
            All processes in the current process group are signaled.

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

Рис. 11: Команда kill

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.