

Презентация по лабораторной работе №7

Маслова А.П.

10 марта 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Маслова Анна Павловна
- студентка, НПИбд-02-23
- факультет физико-математических и естественных наук, направление прикладная информатика
- Российский университет дружбы народов
- 1132236134@pfur.ru

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.

Выполнение лабораторной работы

Выполните все примеры, приведённые в первой части описания лабораторной работы



```
root@kali:~# touch test.txt
root@kali:~# cat test.txt
root@kali:~# echo test.txt
root@kali:~# cat test.txt
root@kali:~# echo test.txt
root@kali:~#
```

Рис. 1: Первая часть лр

Файл equipment

Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.

[illegible]

Рис. 2: Файл equipment

Создание директории ~/ski.places

В домашнем каталоге создайте директорию ~/ski.places

```
quasiro@quasiro ~$ mkdir ~/ski.places
quasiro@quasiro ~$ ls
Desktop  Downloads  Desktop  git-extensions  LICENSE  pom.xml  ski.places  test.txt  work  work  Desktop
```

Рис. 3: Создание директории ~/ski.places

Перемещение файла equipment

Переместите файл equipment в каталог ~/ski.places

```
osmar@osmar:~$ mv equipment ski.places
osmar@osmar:~$ ls
bin  boot  dev  etc  home  lib  lib64  media  mnt  opt  root  sbin  srv  sys  usr  var
osmar@osmar:~$ ls ski.places
equipment
osmar@osmar:~$
```

Рис. 4: Перемещение файла equipment

Переименование файла equipment

Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`

A terminal window with a dark background. The prompt is `qwasir@qwasir ~`. The command `mv ~/ski.places/equipment ~/ski.places/equiplist` has been entered and executed. The prompt is now `qwasir@qwasir ~` followed by a cursor.

```
qwasir@qwasir ~$ mv ~/ski.places/equipment ~/ski.places/equiplist
qwasir@qwasir ~$
qwasir@qwasir ~$
```

Рис. 5: Переименование файла equipment


Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.places, назовите его equiplist2.

```
qemuired@qemuired ~$ touch abc1
qemuired@qemuired ~$ ls
abc  desktop  Downloads  git-reloaded  LICENSE  pass.txt  ski_places  test.txt  work  zzzzzzz
qemuired@qemuired ~$ cp abc1 ski_places/equiplist2
qemuired@qemuired ~$ ls ski_places
equiplist2 equiplist
qemuired@qemuired ~$
```

Рис. 6: Создание abc1

Создание каталога с именем equipment в каталоге ~/ski.places.

Создайте каталог с именем equipment в каталоге ~/ski.places.



```
equinoxe@equinox ~$ mkdir -p ~/ski.places/equipment
equinoxe@equinox ~$ cd ~/ski.places
equinoxe@equinox ~/ski.places$ ls
equinoxe@equinox ~/ski.places$
```

Рис. 7: Создание каталога с именем equipment в каталоге ~/ski.places.

Перемещение файлов ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment

Переместите файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.

```
~/ski.plases$ mv equiplist ~/ski.plases/equipment
~/ski.plases$ mv equiplist2 ~/ski.plases/equipment
~/ski.plases$ ls -la ~/ski.plases/equipment
total 4
-rw-rw-r-- 1 user user 0 Nov 11 11:11 equiplist
-rw-rw-r-- 1 user user 0 Nov 11 11:11 equiplist2
```

Рис. 8: Перемещение файлов ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment

Создайте и переместите каталог ~/newdir в каталог ~/ski.places

Создайте и переместите каталог ~/newdir в каталог ~/ski.places и назовите его plans.

```
gsm@ircodkms100 ~$ mkdir newdir
gsm@ircodkms100 ~$ mv newdir ski.places/plans
gsm@ircodkms100 ~$ ls
dot  Desktop  Downloads  git-repos  LICENSE  pass.txt  ski.places  test.txt  work  zaps  zapshots
gsm@ircodkms100 ~$ ls ski.places
newdir  plans
gsm@ircodkms100 ~$
```

Рис. 9: Создайте и переместите каталог ~/newdir в каталог ~/ski.places

Определение прав доступа

Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

`drwxr--r--` `australia`

`drwx--x--x` `play`

`-r-xr--r--` `my_os`

`-rw-rw-r--` `feathers`

При необходимости создайте нужные файлы.

```
quail@quail:~$ mkdir australia play
quail@quail:~$ ls
abc!  australia  Desktop  Downloads  git-extended  LICENSE  pass.txt  play  ski_places  test.txt  work
quail@quail:~$ touch xy.txt
quail@quail:~$ touch feathers
quail@quail:~$ ls
abc!  australia  Desktop  Downloads  feathers  git-extended  LICENSE  xy.txt  pass.txt  play  ski_places
```

Рис. 10: Создание файлов

Определение прав доступа

```
qpsalm@qpsalm:~$ ls -l -d australia
drwx-r-x. 1 qpsalm qpsalm 0 sep 19 21:42 australia
qpsalm@qpsalm:~$ chmod 744 australia
qpsalm@qpsalm:~$ ls -l -d australia
drwxr-xr-x. 1 qpsalm qpsalm 0 sep 19 21:42 australia
qpsalm@qpsalm:~$ ls -l -d play
drwx-r-x. 1 qpsalm qpsalm 0 sep 19 21:42 play
qpsalm@qpsalm:~$ chmod 777 -r -r play
chmod: cannot perm: 'r-r'
ls -lsdls /home/qpsalm --help покажет подробную информацию.
qpsalm@qpsalm:~$ chmod -d g-r, r-r play
chmod: cannot stat: '-d'
ls -lsdls /home/qpsalm --help покажет подробную информацию.
qpsalm@qpsalm:~$ chmod 722 play
qpsalm@qpsalm:~$ ls -l -d play
drwx-r-x. 1 qpsalm qpsalm 0 sep 19 21:42 play
qpsalm@qpsalm:~$ chmod 777 play
qpsalm@qpsalm:~$ ls -l -d play
drwxr-xr-x. 1 qpsalm qpsalm 0 sep 19 21:42 play
qpsalm@qpsalm:~$ ls -l -d play
ls:ls:
qpsalm@qpsalm:~$ ls -l -d sy.as
-rw-r--r--. 1 qpsalm qpsalm 0 sep 19 21:42 sy.as
qpsalm@qpsalm:~$ ls -lsdls /home/qpsalm --help покажет подробную информацию.
qpsalm@qpsalm:~$ chmod 744 sy.as
qpsalm@qpsalm:~$ ls -l -d sy.as
-rw-r--r--. 1 qpsalm qpsalm 0 sep 19 21:42 sy.as
qpsalm@qpsalm:~$ chmod 744 sy.as
qpsalm@qpsalm:~$ ls -l -d sy.as
-rw-r--r--. 1 qpsalm qpsalm 0 sep 19 21:42 sy.as
qpsalm@qpsalm:~$ ls -l -d feathers
-rw-r--r--. 1 qpsalm qpsalm 0 sep 19 21:43 feathers
qpsalm@qpsalm:~$ chmod 744 feathers
qpsalm@qpsalm:~$ ls -l -d feathers
-rw-r--r--. 1 qpsalm qpsalm 0 sep 19 21:43 feathers
qpsalm@qpsalm:~$
```

Рис. 11: Определение прав доступа

Посмотрите содержимое файла /etc/passwd.



```
qwe@qwe:~$ cat /etc/passwd
cat: /etc/passwd: Перевод файла или каталога
qwe@qwe:~$
```

Рис. 12: Просмотр содержимого password

Скопируем файл ~/feathers в файл ~/file.old

Скопируйте файл ~/feathers в файл ~/file.old.

```
quail@quail:~$ cp feathers file.old
quail@quail:~$ ls
dell  desktop  feathers  git-repos  mp3s  play  test.txt  temp  zanyam  zanyam
mozilla  quail$  /usr/bin  /usr/bin  /usr/bin  /usr/bin  /usr/bin  /usr/bin  /usr/bin  /usr/bin
```

Рис. 13: Скопируем файл ~/feathers в файл ~/file.old

Переместим файл ~/file.old в каталог ~/play

Переместите файл ~/file.old в каталог ~/play.



```
qwe@qwe:~$ mv ~/file.old ~/play
qwe@qwe:~$ mv ~/file.old ~/play
qwe@qwe:~$
```

Рис. 14: Переместим файл ~/file.old в каталог ~/play

Скопируем каталог ~/play в каталог ~/fun

Скопируйте каталог ~/play в каталог ~/fun.

A terminal window showing the contents of the ~/fun directory. The prompt is [qwe@redhatlinux ~]#. The command ls -l ~/fun is entered. The output shows a list of files and directories: dot, debian, fun, howto, fun, git-extended, LICENSE, my.py, my.txt, play, and sub.pl. The file fun is highlighted in green.

```
[qwe@redhatlinux ~]# ls -l ~/fun
[qwe@redhatlinux ~]# ls
dot  debian  fun  howto  fun  git-extended  LICENSE  my.py  my.txt  play  sub.pl
```

Рис. 15: Скопируем каталог ~/play в каталог ~/fun

Перемещение каталога ~/fun в каталог ~/play

Переместите каталог ~/fun в каталог ~/play и назовите его games.

```
osmelirod@osmeliro ~$ mv fun play/games
osmelirod@osmeliro ~$ ls
dot  desktop  fun/games  Downloads  Picturs  git-extended  LICENSE  my.cnf  past.txt  play  xit_games
osmelirod@osmeliro ~$ ls play
games
osmelirod@osmeliro ~$
```

Рис. 16: Перемещение каталога ~/fun в каталог ~/play


Лишение владельца файла ~/feathers права на чтение

Лишите владельца файла ~/feathers права на чтение.

```
osmail@osmail:~$ ls -l feathers  
-rwxr-xr-x 1 osmail osmail 0 sep 19 21:43 feathers  
osmail@osmail:~$ chmod u-r feathers  
osmail@osmail:~$ ls -l feathers  
-rwxr-xr-x 1 osmail osmail 0 sep 19 21:43 feathers  
osmail@osmail:~$
```

Рис. 17: Лишение владельца файла ~/feathers права на чтение

Попытаемся посмотреть файл ~/feathers командой cat и скопировать файл



```
qemir@qemir:~$ cat ~/feathers  
cat: ~/feathers: Permission denied  
qemir@qemir:~$ cat ~/feathers  
cat: ~/feathers: Permission denied  
qemir@qemir:~$ cat ~/feathers  
cat: ~/feathers: Permission denied  
qemir@qemir:~$
```

Рис. 18: Команда cat и ср

Дадим владельцу файла ~/feathers право на чтение

Дайте владельцу файла ~/feathers право на чтение.



```
osmalin@osmalin:~$ chmod u+r feathers
osmalin@osmalin:~$ ls -l feathers
-rw-r--r-- 1 osmalin osmalin 6 May 19 21:43 feathers
osmalin@osmalin:~$
```

Рис. 19: Дадим владельцу файла ~/feathers право на чтение

Переход в каталог ~/play

Перейдите в каталог ~/play. Дайте владельцу каталога ~/play право на выполнение.



```
[kvm@localhost ~]$ cd play
kvm: cd: play: Permission denied
[kvm@localhost ~]$ chmod u+x play
[kvm@localhost ~]$ ls -l -d play
drwxr-xr-x. 1 kvm:kvm 4096 Sep 19 22:00 play
[kvm@localhost ~]$
```

Рис. 21: Переход в каталог ~/play

Прочитайте `man` по командам `mount`, `fsck`, `mkfs`, `kill`.

```

root
kali(R)
AIME
mount -mount a filesystem

SYNOPSIS
mount [-h|-W]
mount [-t] [-t filesystem]
mount -o [-fforce] [-t filesystem] [-o options]
mount -t filesystem [-o options] device mountpoint
mount -t filesystem [-t filesystem] [-o options] device mountpoint
mount -bind(-rbind)[-newfs] source target
mount -make [shared|slave|private|bindable|shared|relaxed|private|unbindable] mountpoint

DESCRIPTION
All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /. These files
the big file tree. Conversely, the mount(8) command will detach it again. The filesystem is used to control a
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.
necessary to mount a filesystem by default. See section "Non-supervisor mounts" below for more details. The pre
the pathname dir refers to the root of the filesystem on device.

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

mount /dev

then mount looks for a mountpoint (and if not found then for a device) in the /etc/passwd file. It's possible to
mount -target mountpoint

The same filesystem may be mounted more than once, and in some cases (e.g., network filesystems) the same file
control this behavior. All behavior is controlled by the kernel and it is usually specific to the filesystem
details.

man mount(8) line 3 (press q to quit)

```

Рис. 22: man mount

```
fsck
fsck(8)                                     System Admin

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-badnrtv] [-o [opt]] [-C [ctl]] [-t fsflag] [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/sda1 or /dev/sda1-980-0123456789) or LABEL=rcv1. Normally, the fsck program will try to handle filesy
    If no filesystems are specified on the command line, and the -A option is not specified, fsck will default to
    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
    32     Checking canceled by user request
    128    Shared-library error

    The exit status returned when multiple filesystems are checked is the bit-wise OR of the exit statuses for each
    In actuality, fsck is simply a forwarder for the various filesystem checkers (fsck.*flags) available under Lin
    Manual page fsck(8) line 1 (press h for help or q to quit)
```

Рис. 23: man fsck

```
foot
dell@s)                                     System Admin

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> utilities.

mkfs is used to build a linux filesystem on a device, usually a hard disk partition. The device argument is an 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.<format>) available under 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 assumed.
    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 is equivalent to specifying --type ext2.
-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 be ignored.)

BUGS
All generic options must precede and not be combined with filesystem-specific options. Some filesystem-specific options may be incompatible with other options.

AUTHORS
David Noy <noy@cs.cmu.edu>, Fred N. van Kempen <van.kempen@tcl.mcgill.ca>, Ron Scoville <scoville@cs.cmu.edu>
The manual page was shamelessly adapted from Ron Scoville's version for the ext2 filesystem.

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

Рис. 24: man mkfs

```
foot
kill()
User Co

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-t timeout milliseconds signal] [-i] pidname...
    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 pro-
    cess. For the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process
    signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before
    Most modern shells have a builtin kill command, with a usage rather similar to that of the command described in
    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
            All processes in the current process group are signaled.

        -1
            All processes with a PID larger than 1 are signaled.

        -n
            where n is larger than 1. All processes in process group n are signaled. When an argument of the form
            must be preceded by a '-' option, otherwise it will be taken as the signal to send.

NAME
    All arguments listed under this name will be signaled.
Manual page kill(1) line 1 (press h for help or q to quit)
```

Рис. 25: man kill

Выводы

Ознакомились с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобрели практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.