

Отчёт по лабораторной работе №6

Дисциплина: Операционные системы

Верниковская Екатерина Андреевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Ответы на контрольные вопросы	17
5	Выводы	20
6	Список литературы	21

Список иллюстраций

3.1	Имя домашнего каталога	7
3.2	Переход в каталог /tmp	7
3.3	Содержимое каталога /tmp	7
3.4	ls -l	8
3.5	ls -a	8
3.6	ls -F	9
3.7	ls -alF	9
3.8	Содержимое каталога /var/spool	9
3.9	Владелец файлов и подкаталогов	10
3.10	Создание каталога newdir	10
3.11	Создание подкаталога morefun	10
3.12	Создание трёх новых подкаталогов	10
3.13	Удаление трёх новых подкаталогов	11
3.14	Попытка удалить ~/newdir	11
3.15	Создание каталога ~/newdir/morefun	11
3.16	Команда man ls	11
3.17	Опция -R	12
3.18	Опция -lt	12
3.19	Использование -R	12
3.20	Использование -lt	12
3.21	Основные команды cd	13
3.22	Основные команды pwd	13
3.23	Основные команды mkdir	13
3.24	Основные команды rmdir	14
3.25	Основные команды rm	14
3.26	История наших команд	15
3.27	Модификация команды (1)	15
3.28	Модификация команды (2)	16
3.29	Модификация команды (3)	16

Список таблиц

1 Цель работы

Приобрести практические навыки взаимодействия пользователя с системой посредством командной строки.

2 Задание

1. Определить полное имя домашнего каталога.
2. Перейти в каталог `/tmp` и вывести содержимое каталога командой `ls` с различными опциями.
3. Определить есть ли в каталоге `/var/spool` подкаталог с именем `cron`.
4. Определить кто является владельцем файлов и подкаталогов в домашнем каталоге.
5. Создать каталог с именем `newdir` и проделать в нём разные действия (создать каталоги, удалить и т.д.)
6. Научиться пользоваться командой `man`.
7. Научиться пользоваться командой `history`.

3 Выполнение лабораторной работы

С помощью команды *pwd* определяем полное имя нашего домашнего каталога (рис. 3.1)

```
[eavernikovskaya@eavernikovskaya ~]$ pwd
/home/eavernikovskaya
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.1: Имя домашнего каталога

С помощью команды *cd* переходим в каталог */tmp* (рис. 3.2)

```
[eavernikovskaya@eavernikovskaya ~]$ cd /tmp
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.2: Переход в каталог */tmp*

Выводим содержимое каталога */tmp* с помощью команды *ls* (рис. 3.3)

```
[eavernikovskaya@eavernikovskaya tmp]$ ls
sddm-auth-941e1ea1-4838-417d-bcee-1f32b6ef34cf
sddm--sGjbPh
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-chrond.service-TtnVal
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-dbus-broker.service-XBzLRa
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-ModemManager.service-lcX4h2
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-polkit.service-r7thQu
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-rtkit-daemon.service-a3XNbd
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-logind.service-5Jowqu
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-oomd.service-RowNw0
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-resolved.service-2DA89H
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-upower.service-Dswagx
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.3: Содержимое каталога */tmp*

Далее мы используем команду `ls` с различными опциями. Это `ls -l` (отображает подробный список файлов и папок с дополнительной информацией, такой как права доступа к файлу, владелец, группа, размер файла, время последнего изменения и имя файла.), `ls -a` (выводит все файлы и папки, включая скрытые файлы, которые начинаются с точки), `ls -F` (добавляет символ на конец каждого выводимого имени файла или папки для обозначения их типа) и `ls -alF` (Комбинация опций `-l`, `-a`, `-F`) (рис. 3.4), (рис. 3.5), (рис. 3.6), (рис. 3.7)

```
[eavernikovskaya@eavernikovskaya tmp]$ ls -l
итого 0
srwxr-xr-x. 1 root root 0 map 12 18:33 sddm-auth-941e1ea1-4838-417d-bcee-1f32b6ef34cf
srwx----- 1 sddm sddm 0 map 12 18:33 sddm--sGjbPh
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-chrond.service-TtnVal
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-dbus-broker.service-XBzLRa
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-ModemManager.service-lcX4h2
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-polkit.service-r7thQu
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-rtkit-daemon.service-a3XNbD
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-logind.service-5JoWqu
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-oomd.service-RowNw0
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-resolved.service-2DA89H
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-upower.service-Dswagx
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.4: `ls -l`

```
[eavernikovskaya@eavernikovskaya tmp]$ ls -a
.
..
.font-unix
.ICE-unix
sddm-auth-941e1ea1-4838-417d-bcee-1f32b6ef34cf
sddm--sGjbPh
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-chrond.service-TtnVal
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-dbus-broker.service-XBzLRa
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-ModemManager.service-lcX4h2
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-polkit.service-r7thQu
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-rtkit-daemon.service-a3XNbD
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-logind.service-5JoWqu
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-oomd.service-RowNw0
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-resolved.service-2DA89H
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-upower.service-Dswagx
.X0-lock
.X11-unix
.XIM-unix
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.5: `ls -a`


```
[eavernikovskaya@eavernikovskaya tmp]$ ls -F
sddm-auth-941e1ea1-4838-417d-bcee-1f32b6ef34cf=
sddm--s6jbPh=
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-chronyd.service-TtnVal/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-dbus-broker.service-XBzLRa/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-ModemManager.service-lcX4h2/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-polkit.service-r7thQu/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-rtkit-daemon.service-a3XNbd/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-logind.service-5JoWqu/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-oemd.service-RowNw0/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-resolved.service-2DA89H/
systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-upower.service-Dswagx/
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.6: ls -F

```
[eavernikovskaya@eavernikovskaya tmp]$ ls -alF
total 4
drwxrwxrwt. 15 root root 360 map 12 18:36 ./
dr-xr-xr-x. 1 root root 158 hor 1 04:05 ../
drwxrwxrwt. 2 root root 40 map 12 18:33 front-unix/
drwxrwxrwt. 2 root root 40 map 12 18:33 lcf-unix/
srwxr-xr-x. 1 root root 0 map 12 18:33 sddm-auth-941e1ea1-4838-417d-bcee-1f32b6ef34cf=
srwx----- 1 sddm sddm 0 map 12 18:33 sddm--s6jbPh=
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-chronyd.servic
e-TtnVal/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-dbus-broker.se
rvice-XBzLRa/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-ModemManager.s
ervice-lcX4h2/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-polkit.service
-r7thQu/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-rtkit-daemon.s
ervice-a3XNbd/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-logind
.service-5JoWqu/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-oemd.s
ervice-RowNw0/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-systemd-resolv
ed.service-2DA89H/
drwx----- 3 root root 60 map 12 18:33 systemd-private-d57a24b3f3a24144a59cdf98cd25ec40-upower.service
-Dswagx/
-r--r--r-- 1 eavernikovskaya eavernikovskaya 11 map 12 18:33 .X0-lock
drwxrwxrwt. 2 root root 60 map 12 18:33 .X11-unix/
drwxrwxrwt. 2 root root 40 map 12 18:33 .XIM-unix/
[eavernikovskaya@eavernikovskaya tmp]$
```

Рис. 3.7: ls -alF

Переходим в каталог `/var/spool` и с помощью команды `ls` определяем есть ли там подкаталог с именем `cron`. Подкаталог `cron` есть в каталоге `/var/spool` (рис. 3.8)

```
[eavernikovskaya@eavernikovskaya tmp]$ cd /var/spool
[eavernikovskaya@eavernikovskaya spool]$ ls
abrt abrt-upload anacron at cron cups lpd mail plymouth
[eavernikovskaya@eavernikovskaya spool]$
```

Рис. 3.8: Содержимое каталога `/var/spool`

Переходим в домашний каталог и с помощью команды `ls -l` выводим содержимое каталога и узнаём кто является владельцем файлов и подкаталогов (рис. 3.9)

```
[eavernikovskaya@eavernikovskaya spool]$ cd
[eavernikovskaya@eavernikovskaya ~]$ ls -l
итого 20
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 132 мар 3 21:29 git-extended
-rw-r--r--. 1 eavernikovskaya eavernikovskaya 18657 мар 12 14:17 LICENSE
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 Видео
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 Документы
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 90 фев 20 20:11 Загрузки
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 156 мар 12 13:47 Изображения
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 Музыка
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 Общедоступные
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 'Рабочий стол'
drwxr-xr-x. 1 eavernikovskaya eavernikovskaya 0 фев 20 19:06 Шаблоны
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.9: Владелец файлов и подкаталогов

В домашнем каталоге создаём каталог с именем `newdir` (рис. 3.10)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir newdir
[eavernikovskaya@eavernikovskaya ~]$ ls
git-extended  newdir  Документы  Изображения  Общедоступные  Шаблоны
LICENSE       Видео  Загрузки  Музыка      'Рабочий стол'
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.10: Создание каталога `newdir`

В каталоге `~/newdir` создаём новый каталог с именем `morefun` (рис. 3.11)

```
[eavernikovskaya@eavernikovskaya ~]$ mkdir ~/newdir/morefun
[eavernikovskaya@eavernikovskaya ~]$ cd ~/newdir/
[eavernikovskaya@eavernikovskaya newdir]$ ls
morefun
[eavernikovskaya@eavernikovskaya newdir]$
```

Рис. 3.11: Создание подкаталога `morefun`

Далее в домашнем каталоге создаём одной командой `mkdir` три новых каталога с именами `letters`, `memos`, `misk` (рис. 3.12)

```
[eavernikovskaya@eavernikovskaya newdir]$ cd
[eavernikovskaya@eavernikovskaya ~]$ mkdir letters memos misk
[eavernikovskaya@eavernikovskaya ~]$ ls
git-extended  LICENSE  misk  Видео  Загрузки  Музыка  'Рабочий стол'
letters       memos   newdir  Документы  Изображения  Общедоступные  Шаблоны
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.12: Создание трёх новых подкаталогов

Затем удаляем их одной командой `rmdir -p` (рис. 3.13)

```
[eavernikovskaya@eavernikovskaya ~]$ rmdir -p letters/ memos/ misk/
[eavernikovskaya@eavernikovskaya ~]$ ls
git-extended  newdir  Документы  Изображения  Общедоступные  Шаблоны
LICENSE       Видео   Загрузки   Музыка        'Рабочий стол'
```

Рис. 3.13: Удаление трёх новых подкаталогов

Далее пробуем удалить ранее созданный каталог `~/newdir` командой `rm`. Но у нас не получится это сделать, так как `rm` удаляет файлы, а не каталоги (рис. 3.14)

```
[eavernikovskaya@eavernikovskaya ~]$ rm ~/newdir/
rm: невозможно удалить '/home/eavernikovskaya/newdir/': Это каталог
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.14: Попытка удалить `~/newdir`

После удаляем каталог `~/newdir/morefun` из домашнего каталога командой `rmdir -p`. А затем проверяем, был ли каталог удалён (рис. 3.15)

```
[eavernikovskaya@eavernikovskaya ~]$ rmdir -p newdir/morefun
[eavernikovskaya@eavernikovskaya ~]$ ls
git-extended  LICENSE  Видео   Документы  Загрузки  Изображения  Музыка  Общедоступные  'Рабочий стол'  Шаблоны
```

Рис. 3.15: Создание каталога `~/newdir/morefun`

- Далее с помощью команды `man` определяем, какую опцию команды `ls` нужно использовать для просмотра содержимого не только указанного каталога, но и подкаталогов, входящих в него (это `-R`)
- После с помощью команды `man` определяем набор опций команды `ls`, позволяющий отсортировать по времени последнего изменения выводимый список содержимого каталога с развёрнутым описанием файлов (это `-lt`) (рис. 3.16), (рис. 3.17), (рис. 3.18), (рис. 3.19), (рис. 3.20)

```
[eavernikovskaya@eavernikovskaya ~]$ man ls
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.16: Команда `man ls`

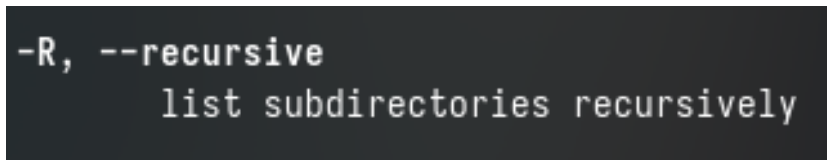


Рис. 3.17: Опция -R

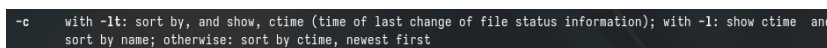


Рис. 3.18: Опция -lt

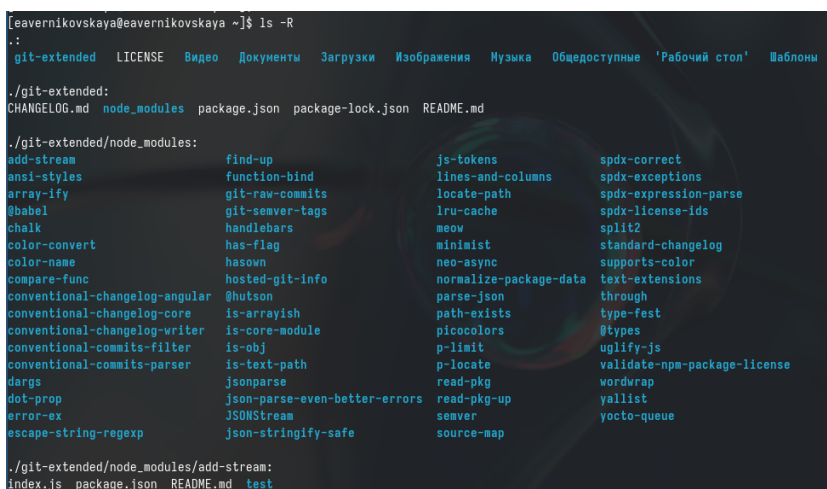


Рис. 3.19: Использование -R

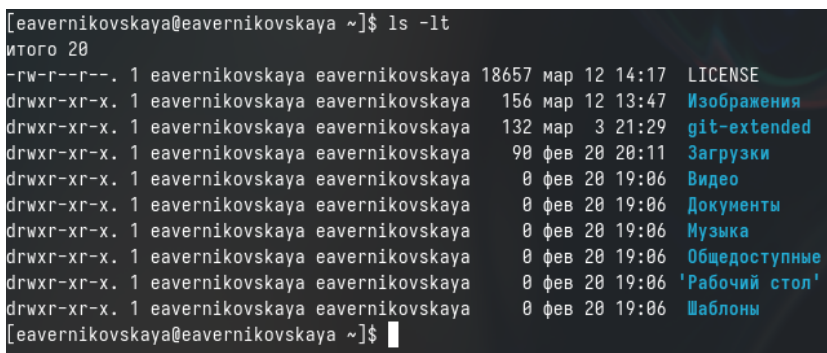


Рис. 3.20: Использование -lt

Далее используем команду *man* для просмотра описания следующих команд: *cd*, *pwd*, *mkdir*, *rmdir*, *rm*. Просматриваем основные опции этих команд (рис. 3.21), (рис. 3.22), (рис. 3.23), (рис. 3.24), (рис. 3.25)

```

[eaevernikovskaya@eaevernikovskaya ~]$ man bash | grep "cd"
troff:<standard input>:2881: warning: cannot select font 'CW'
OLDPWD The previous working directory as set by the cd command.
PWD The current working directory as set by the cd command.
CDPATH The search path for the cd command. This is a colon-separated list of directories in which the shell looks
for destination directories specified by the cd command. A sample value is ".:~/usr".
HOME The home directory of the current user; the default argument for the cd builtin command. The value of this
troff:<standard input>:3724: warning: cannot select font 'CW'
troff:<standard input>:3724: warning: cannot select font 'CW'
range expressions, where [a-d] is equivalent to [abcd], set value of the LC_ALL shell variable to C,
troff:<standard input>:5391: warning: cannot select font 'CW'
troff:<standard input>:5394: warning: cannot select font 'CW'
• the current working directory as set by cd, pushd, or popd, or inherited by the shell at invocation
troff:<standard input>:5978: warning: cannot select font 'CW'
troff:<standard input>:5990: warning: cannot select font 'CW'
troff:<standard input>:7250: warning: cannot select font 'CW'
the cdspell shell option. Word boundaries are the same as those used by shell-forward-word.
within the function. The cd builtin is commonly redefined this way. The return status is false if
cd [-l[[-P [-e]] [-@]] [dir]

```

Рис. 3.21: Основные команды cd

```

PWD(1) User Commands PWD(1)
NAME
  pwd - print name of current/working directory
SYNOPSIS
  pwd [OPTION]...
DESCRIPTION
  Print the full filename of the current working directory.

  -l, --logical
      use PWD from environment, even if it contains symlinks

  -P, --physical
      avoid all symlinks

  --help display this help and exit

  --version
      output version information and exit

  If no option is specified, -P is assumed.

  NOTE: your shell may have its own version of pwd, which usually supersedes the version described here. Please refer
  to your shell's documentation for details about the options it supports.

```

Рис. 3.22: Основные команды pwd

```

MKDIR(1) User Commands MKDIR(1)
NAME
  mkdir - make directories
SYNOPSIS
  mkdir [OPTION]... DIRECTORY...
DESCRIPTION
  Create the DIRECTORY(ies), if they do not already exist.

  Mandatory arguments to long options are mandatory for short options too.

  -m, --mode=MODE
      set file mode (as in chmod), not a=rwx - umask

  -p, --parents
      no error if existing, make parent directories as needed, with their file modes unaffected by any -m option.

  -v, --verbose
      print a message for each created directory

  -Z
      set SELinux security context of each created directory to the default type

  --context[=CTX]
      like -Z, or if CTX is specified then set the SELinux or SMACK security context to CTX

  --help display this help and exit

  --version
      output version information and exit

```

Рис. 3.23: Основные команды mkdir

```
RMDIR(1) User Commands RMDIR(1)
NAME
  rmdir - remove empty directories
SYNOPSIS
  rmdir [OPTION]... DIRECTORY...
DESCRIPTION
  Remove the DIRECTORY(ies), if they are empty.
  --ignore-fail-on-non-empty
    ignore each failure to remove a non-empty directory
  -p, --parents
    remove DIRECTORY and its ancestors; e.g., 'rmdir -p a/b' is similar to 'rmdir a/b a'
  -v, --verbose
    output a diagnostic for every directory processed
  --help
    display this help and exit
  --version
    output version information and exit
```

Рис. 3.24: Основные команды rmdir

```
RM(1) User Commands RM(1)
NAME
  rm - remove files or directories
SYNOPSIS
  rm [OPTION]... [FILE]...
DESCRIPTION
  This manual page documents the GNU version of rm. rm removes each specified file. By default, it does not remove directories.
  If the -I or --interactive=once option is given, and there are more than three files or the -r, -B, or --recursive are given, then rm prompts the user for whether to proceed with the entire operation. If the response is not affirmative, the entire command is aborted.
  Otherwise, if a file is unwritable, standard input is a terminal, and the -f or --force option is not given, or the -i or --interactive=always option is given, rm prompts the user for whether to remove the file. If the response is not affirmative, the file is skipped.
OPTIONS
  Remove (unlink) the FILE(s).
  -f, --force
    ignore nonexistent files and arguments, never prompt
  -i
    prompt before every removal
  -I
    prompt once before removing more than three files, or when removing recursively; less intrusive than -i, while still giving protection against most mistakes
  --interactive[=WHEN]
    prompt according to WHEN: never, once (-I), or always (-i); without WHEN, prompt always
  --one-file-system
    when removing a hierarchy recursively, skip any directory that is on a file system different from that of the corresponding command line argument
  --no-preserve-root
    do not treat '/' specially
  --preserve-root[=all]
    do not remove '/' (default); with 'all', reject any command line argument on a separate device from its parent
  -r, -R, --recursive
    remove directories and their contents recursively
  -d, --dir
    remove empty directories
  -v, --verbose
    explain what is being done
  --help
    display this help and exit
```

Рис. 3.25: Основные команды rm

С помощью команды *history* просматриваем историю наших команд (рис. 3.26)

```

[eavernikovskaya@eavernikovskaya ~]$ history
 1  sudo -i
 2  tmux
 3  mc
 4  sudo -i
 5  mc
 6  sudo -i
 7  ыгвщ -ш
 8  sudo -i
 9  cd ~/Загрузки/
10  ls
11  tar -xvf pandoc-crossref-Linux.tar.xz
12  ls
13  mv pandoc-crossref /usr/local/bin
14  sudo mv pandoc-crossref /usr/local/bin
15  exit
16  xclip -i < ~/.ssh/id_ed25519.pub
17  ssh-keygen -t ed25519
18  xclip -i < ~/.ssh/id_ed25519.pub
19  gpg --full-generate-key
20  gpg --list-secret-keys --keyid-format LONG
21  gpg --armor --export 47519BDC62A55FE7

```

Рис. 3.26: История наших команд

Далее выполняем модификацию и исполнение нескольких команд из буфера команд введя при этом *!номер_команды:s/что_меняем/на_что_меняем* (рис. 3.27), (рис. 3.28), (рис. 3.29)

```

[eavernikovskaya@eavernikovskaya ~]$ !113:s/newdir/katerok
mkdir katerok
[eavernikovskaya@eavernikovskaya ~]$ ls
git-extended  LICENSE  Документы  Изображения  Общедоступные  Шаблоны
katerok       Видео   Загрузки   Музыка        'Рабочий стол'
[eavernikovskaya@eavernikovskaya ~]$

```

Рис. 3.27: Модификация команды (1)

```
[eavernikovskaya@eavernikovskaya ~]$ !108:s/rm/pwd
pwd --help
pwd: pwd [-LP]
    Print the name of the current working directory.

Options:
  -L      print the value of $PWD if it names the current working
          directory
  -P      print the physical directory, without any symbolic links

By default, `pwd' behaves as if `-L' were specified.

Exit Status:
Returns 0 unless an invalid option is given or the current directory
cannot be read.
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.28: Модификация команды (2)

```
[eavernikovskaya@eavernikovskaya ~]$ !122:s/R/F
ls -F
git-extended/  LICENSE  Документы/  Изображения/  Общедоступные/  Шаблоны/
katerok/      Видео/   Загрузки/   Музыка/        'Рабочий стол'/
[eavernikovskaya@eavernikovskaya ~]$
```

Рис. 3.29: Модификация команды (3)

4 Ответы на контрольные вопросы

1. Что такое командная строка?

Командная строка - это текстовая система, которая передает команды компьютеру и возвращает результаты пользователю. В операционной системе типа Linux взаимодействие пользователя с системой обычно осуществляется с помощью командной строки посредством построчного ввода команд.

2. При помощи какой команды можно определить абсолютный путь текущего каталога? Приведите пример.

Для определения абсолютного пути к текущему каталогу используется команда `pwd`. Например: если я введу `pwd` в своем домашнем каталоге то получу `/home/eavernikovskaya`

3. При помощи какой команды и каких опций можно определить только тип файлов и их имена в текущем каталоге? Приведите примеры.

С помощью команды `ls` можно определить имена файлов, при помощи опции `-F` уже мы сможем определить тип файлов, если нам необходимы скрытые файлы, добавим опцию `-a`. Пример есть в лабораторной работе

4. Каким образом отобразить информацию о скрытых файлах? Приведите примеры.

С помощью команды `ls` можно определить имена файлов, если нам необходимы скрытые файлы, добавим опцию `-a`. Пример есть в лабораторной работе.

5. При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой? Приведите примеры.

`rmdir` по умолчанию удаляет пустые каталоги, не удаляет файлы. `rm` удаляет файлы, без дополнительных опций (`-d`, `-r`) не будет удалять каталоги. Удалить в одной строчке одной командой можно файл и каталог. Если файл находится в каталоге, используем рекурсивное удаление, если файл и каталог не связаны подобным образом, то добавим опцию `-d`, введя имена через пробел после утилиты.

6. Каким образом можно вывести информацию о последних выполненных пользователем командах? работы?

Вывести информацию о последних выполненных пользователем команд можно с помощью `history`. Пример приведет в лабораторной работе.

7. Как воспользоваться историей команд для их модифицированного выполнения? Приведите примеры.

Используем синтаксис `!номеркоманды` в выводе `history:s/что заменяем/на что заменяем`. Примеры приведены в лабораторной работе.

8. Приведите примеры запуска нескольких команд в одной строке.

Предположим, я нахожусь не в домашнем каталоге. Если я введу `"cd ; ls"`, то окажусь в домашнем каталоге и получу вывод файлов внутри него

9. Дайте определение и приведите примера символов экранирования.

Символ экранирования - (обратный слеш) добавление перед спецсимволом обратный слеш, чтобы использовать специальный символ как обычный. Также позволяет читать системе название директорий с пробелом. Пример: `cd work/Операционные системы/`

10. Охарактеризуйте вывод информации на экран после выполнения команды `ls` с опцией `l`.

Опция `-l` позволит увидеть дополнительную информацию о файлах в каталоге: время создания, владельца, права доступа

11. Что такое относительный путь к файлу? Приведите примеры использования относительного и абсолютного пути при выполнении какой-либо команды.

Относительный путь к файлу начинается из той директории, где вы находитесь (она сама не прописывается в пути), он прописывается относительно данной директории. Абсолютный путь начинается с корневого каталога.

12. Как получить информацию об интересующей вас команде?

Использовать `man` или `-help`

13. Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?

Клавиша `Tab`.

5 Выводы

В ходе выполнения лабораторной работы мы приобрели практические навыки взаимодействия пользователя с системой посредством командной строки.

6 Список литературы

Не пользовалась сайтами.