

Отчёт по лабораторной работе № 8

Операционные системы

Зинченко Анастасия Романовна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	8
4	Выводы	20
	Список литературы	21

Список иллюстраций

3.1	Вход	8
3.2	Запись в файл	8
3.3	Названия файлов	9
3.4	Имена всех файлов	10
3.5	Запись в conf.txt	10
3.6	Определение файлов	11
3.7	Имена файлов	11
3.8	Запуск процесса	11
3.9	Удаление файла	12
3.10	Запуск gedit	12
3.11	Определение идентификатора процесса gedit	12
3.12	Справка команды kill	13
3.13	Завершение процесса gedit	13
3.14	Справка df	14
3.15	Справка du	15
3.16	df и du	15
3.17	Справка команды find	16
3.18	Имена всех директорий	17

Список таблиц

1 Цель работы

Ознакомиться с инструментами поиска файлов и фильтрации текстовых данных. Приобрести практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

2 Задание

1. Осуществить вход в систему, используя соответствующее имя пользователя.
2. Записать в файл `file.txt` названия файлов, содержащихся в каталоге `/etc`.
Дописать в этот же файл названия файлов, содержащихся в домашнем каталоге.
3. Вывести имена всех файлов из `file.txt`, имеющих расширение `.conf`, после чего записать их в новый текстовый файл `conf.txt`.
4. Определить, какие файлы в домашнем каталоге имеют имена, начинающиеся с символа `c`?
5. Вывести на экран (по странично) имена файлов из каталога `/etc`, начинающиеся с символа `h`.
6. Запустить в фоновом режиме процесс, который будет записывать в файл `~/logfile` файлы, имена которых начинаются с `log`.
7. Удалить файл `~/logfile`.
8. Запустить из консоли в фоновом режиме редактор `gedit`.
9. Определить идентификатор процесса `gedit`, используя команду `ps`, конвейер и фильтр `grep`.
10. Прочитать справку (`man`) команды `kill`, после чего использовать её для завершения процесса `gedit`.
11. Выполнить команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man`.
12. Воспользовавшись справкой команды `find`, вывести имена всех директорий, имеющихся в домашнем каталоге.

13. Контрольные вопросы.

3 Выполнение лабораторной работы

Осуществитвила вход в систему, используя соответствующее имя пользователя.
(рис. 3.1).

A terminal window with a black background and white text. The prompt is [arzinchenko@arzinchenko ~]\$ followed by a white cursor box.

Рис. 3.1: Вход

Записала в файл file.txt названия файлов, содержащихся в каталоге /etc. (рис. 3.2).

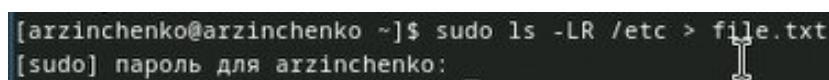
A terminal window with a black background and white text. The first line is [arzinchenko@arzinchenko ~]\$ sudo ls -LR /etc > file.txt. The second line is [sudo] пароль для arzinchenko: followed by a white cursor box.

Рис. 3.2: Запись в файл

Дописала в этот же файл названия файлов, содержащихся в домашнем каталоге.
(рис. 3.3).


```

[arzinchenko@arzinchenko ~]$ ls -lR ~/ > file.txt
ls: невозможно открыть каталог '/home/arzinchenko/play/games': От-
казано в доступе
[arzinchenko@arzinchenko ~]$ head file.txt
/home/arzinchenko/:
итого 0
-rw-rw-r--. 1 arzinchenko arzinchenko  0 map 20 21:38 abc1
drwxr-xr-x. 1 arzinchenko arzinchenko  0 map 15 19:28 Anastasiia
drwxr--r--. 1 arzinchenko arzinchenko  0 map 20 21:43 australia
drwxr-xr-x. 1 arzinchenko arzinchenko 22 map 14 13:12 bin
drwxr-xr-x. 1 arzinchenko arzinchenko 300 map 14 13:13 dotfiles
-rw-rw-r--. 1 arzinchenko arzinchenko  0 map 20 21:48 feathers
-rw-r--r--. 1 arzinchenko arzinchenko  0 map 30 09:47 file.txt
drwxr-xr-x. 1 arzinchenko arzinchenko 74 map  9 14:56 git-extend
ed

```

Рис. 3.3: Названия файлов

Вывела имена всех файлов из file.txt, имеющих расширение .conf. (рис. 3.4).

```

/home/arzinchenko/dotfiles/dot_config/systemd:
/home/arzinchenko/dotfiles/dot_config/systemd/user:
/home/arzinchenko/dotfiles/dot_config/tmux:
-rw-r--r--. 1 arzinchenko arzinchenko 92820 map 14 13:13 tmux.conf
-rw-r--r--. 1 arzinchenko arzinchenko 17230 map 14 13:13 tmux.conf.local
drwxr-xr-x. 1 arzinchenko arzinchenko 16 map 2 20:40 config
-rw-r--r--. 1 arzinchenko arzinchenko 227 map 16 18:10 jsconfig.json
/home/arzinchenko/work/blog/config:
/home/arzinchenko/work/blog/config/_default:
drwxr-xr-x. 1 arzinchenko arzinchenko 96 map 2 20:40 conference-paper
/home/arzinchenko/work/blog/content/publication/conference-paper:
-rw-r--r--. 1 arzinchenko arzinchenko 15043 map 2 20:40 conference-paper.pdf
-rw-r--r--. 1 arzinchenko arzinchenko 653 map 16 18:10 mathja-config.js
drwxr-xr-x. 1 arzinchenko arzinchenko 628 map 2 20:54 conference-paper
/home/arzinchenko/work/blog/public/publication/conference-paper:
-rw-r--r--. 1 arzinchenko arzinchenko 15043 map 16 18:10 conference-paper.pdf
drwxr-xr-x. 1 arzinchenko arzinchenko 46 map 2 20:54 paper-conference
/home/arzinchenko/work/blog/public/publication-type/paper-conference:
/home/arzinchenko/work/blog/public/publication-type/paper-conference/page:
/home/arzinchenko/work/blog/public/publication-type/paper-conference/page/1:
drwxr-xr-x. 1 arzinchenko arzinchenko 1016 map 2 20:40 conference-paper
/home/arzinchenko/work/blog/resources/_gen/images/publication/conference-paper:
drwxr-xr-x. 1 arzinchenko arzinchenko 24 фев 22 12:24 config
/home/arzinchenko/work/study/2023-2024/Операционные системы/os-intro/config:
/home/arzinchenko/work/study/2023-2024/Операционные системы/os-intro/config/course:
/home/arzinchenko/work/study/2023-2024/Операционные системы/os-intro/config/script:

```

Рис. 3.4: Имена всех файлов

Записала их в новый текстовый файл conf.txt. (рис. 3.5).

```

[arzinchenko@arzinchenko ~]$ grep .conf file.txt > conf.txt

```

Рис. 3.5: Запись в conf.txt

Определила, какие файлы в домашнем каталоге имеют имена, начинавшиеся с символа с. (рис. 3.6).

```
[arzinchenko@arzinchenko ~]$ find ~ -name "c*" -print
/home/arzinchenko/.mozilla/firefox/Crash Reports/crashreporter.in
i
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/crash
es
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/compa
tibility.ini
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/cooki
es.sqlite
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/cert9
.db
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/stora
ge/permanent/chrome
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/stora
ge/default/https+++web.telegram.org/cache
/home/arzinchenko/.mozilla/firefox/nefbujxp.default-release/stora
ge/default/https+++web.telegram.org/cache/caches.sqlite
```

Рис. 3.6: Определение файлов

Вывела на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h. (рис. 3.7).

```
[arzinchenko@arzinchenko ~]$ sudo find /etc -name "h*" -print
[sudo] пароль для arzinchenko:
/etc/avahi/hosts
/etc/firewalld/helpers
/etc/libibverbs.d/hfi1verbs.driver
/etc/libibverbs.d/hns.driver
/etc/systemd/homed.conf
/etc/udev/hwdb.d
/etc/udev/hwdb.bin
/etc/host.conf
/etc/hosts
/etc/hostname
/etc/mercurial/hgrc.d
```

Рис. 3.7: Имена файлов

Запустила в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. (рис. 3.8).

```
[arzinchenko@arzinchenko ~]$ find ~ -name "log*" -print > logfile
&
[1] 2690
```

Рис. 3.8: Запуск процесса

Удалила файл ~/logfile. (рис. 3.9).

```

[arzinchenko@arzinchenko ~]$ rm logfile
[arzinchenko@arzinchenko ~]$ ls -l
итого 188
-rw-rw-r--. 1 arzinchenko arzinchenko      0 мар 20 21:38 abc1
drwxr-xr-x. 1 arzinchenko arzinchenko      0 мар 15 19:28 Anastasia
drwxr--r--. 1 arzinchenko arzinchenko      0 мар 20 21:43 australia
drwxr-xr-x. 1 arzinchenko arzinchenko     22 мар 14 13:12 bin
-rw-r--r--. 1 arzinchenko arzinchenko   5395 мар 30 10:06 conf.txt
drwxr-xr-x. 1 arzinchenko arzinchenko     300 мар 14 13:13 dotfiles
-rw-rw-r--. 1 arzinchenko arzinchenko      0 мар 20 21:48 feathers
-rw-r--r--. 1 arzinchenko arzinchenko 178216 мар 30 09:47 file.txt
drwxr-xr-x. 1 arzinchenko arzinchenko      74 мар  9 14:56 git-extended
-rw-r--r--. 1 arzinchenko arzinchenko   2710 мар 30 10:13 loffile
-rw-r--r--. 1 arzinchenko arzinchenko      0 мар 20 20:52 may
drwx--x--x. 1 arzinchenko arzinchenko      0 мар 20 20:57 monthly
-r-xr--r--. 1 arzinchenko arzinchenko      0 мар 20 21:46 my_os
drwx--x--x. 1 arzinchenko arzinchenko     26 мар 20 22:06 play
drwxr-xr-x. 1 arzinchenko arzinchenko     14 мар 20 20:29 reports
drwxr-xr-x. 1 arzinchenko arzinchenko     28 мар 20 21:41 ski.places
drwxr-xr-x. 1 arzinchenko arzinchenko     76 мар  2 20:45 work
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 Видео
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 Документы
drwxr-xr-x. 1 arzinchenko arzinchenko     760 мар 23 18:12 Загрузки
drwxr-xr-x. 1 arzinchenko arzinchenko     858 мар 14 15:47 Изображения
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 Музыка
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 Общедоступные
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 'Рабочий стол'
drwxr-xr-x. 1 arzinchenko arzinchenko      0 фев 15 21:42 Шаблоны

```

Рис. 3.9: Удаление файла

Запустила из консоли в фоновом режиме редактор gedit. (рис. 3.10).

```

[arzinchenko@arzinchenko ~]$ gedit &
[1] 2745

```

Рис. 3.10: Запуск gedit

Определила идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. (рис. 3.11).

```

[arzinchenko@arzinchenko ~]$ ps aux | grep gedit
arzinch+  2745  0.3  1.3 843492 62256 pts/0    Sl   10:19   0:00 gedit
arzinch+  2765  0.0  0.0 222456 2304 pts/0     S+   10:21   0:00 grep --
color=auto gedit
[arzinchenko@arzinchenko ~]$ pgrep gedit
2745
[arzinchenko@arzinchenko ~]$ ps aux | greep gedit | grep -v grep
bash: greep: команда не найдена
[arzinchenko@arzinchenko ~]$ ps aux | grep gedit | grep -v grep
arzinch+  2745  0.1  1.3 843492 62256 pts/0    Sl   10:19   0:00 gedit

```

Рис. 3.11: Определение идентификатора процесса gedit

Прочитала справку (man) команды kill. (рис. 3.12).

```
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.
```

Рис. 3.12: Справка команды kill

Использовала команду kill для завершения процесса gedit. (рис. 3.13).

```
[arzinchenko@arzinchenko ~]$ kill 2745
```

Рис. 3.13: Завершение процесса gedit

Прочитала справку команды df. (рис. 3.14).


```
DE(1)                                User Commands                                DE(1)

NAME
    df - report file system space usage

SYNOPSIS
    df [OPTION]... [FILE]...

DESCRIPTION
    This manual page documents the GNU version of df. df displays the amount
    of space available on the file system containing each file name argument.
    If no file name is given, the space available on all currently mounted
    file systems is shown. Space is shown in 1K blocks by default, unless
    the environment variable POSIXLY_CORRECT is set, in which case 512-byte
    blocks are used.

    If an argument is the absolute file name of a device node containing a
    mounted file system, df shows the space available on that file system
    rather than on the file system containing the device node. This version
    of df cannot show the space available on unmounted file systems, because
    on most kinds of systems doing so requires very nonportable intimate
    knowledge of file system structures.

OPTIONS
    Show information about the file system on which each FILE resides, or all
    file systems by default.

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

    -a, --all
        include pseudo, duplicate, inaccessible file systems

    -B, --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '-BM' prints sizes
        in units of 1,048,576 bytes; see SIZE format below

    --direct
        show statistics for a file instead of mount point

    -h, --human-readable
        print sizes in powers of 1024 (e.g., 1023M)

    -H, --si
```

Рис. 3.14: Справка df

Прочитала справку команды du. (рис. 3.15).

```

DU(1)                                     User Commands                                     DU(1)

NAME
    du - estimate file space usage

SYNOPSIS
    du [OPTION]... [FILE]...
    du [OPTION]... --files0-from=F

DESCRIPTION
    Summarize device usage of the set of FILES, recursively for directories.

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

    -0, --null
        end each output line with NUL, not newline

    -a, --all
        write counts for all files, not just directories

    --apparent-size
        print apparent sizes rather than device usage; although the appar-
        ent size is usually smaller, it may be larger due to holes in
        ('sparse') files, internal fragmentation, indirect blocks, and the
        like

    -B, --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '-BM' prints sizes
        in units of 1,048,576 bytes; see SIZE format below

    -b, --bytes
        equivalent to '--apparent-size --block-size=1'

    -c, --total
        produce a grand total

    -D, --dereference-args
        dereference only symlinks that are listed on the command line

    -d, --max-depth=N
        print the total for a directory (or file, with --all) only if it
        is N or fewer levels below the command line argument;
        --max-depth=0 is the same as --summarize

```

Рис. 3.15: Справка du

Выполнила команды df и du. (рис. 3.16).

```

[arzinchenko@arzinchenko ~]$ df -vi
Файловая система  Инодов  ИИспользовано  ИСвободно  ИИспользовано%  Смонтировано в
/dev/sda3          0          0          0          - /
devtmpfs           574558        510      574048          1% /dev
tmpfs              579768         8      579760          1% /dev/shm
tmpfs             819200        852      818348          1% /run
tmpfs            1048576         31     1048545          1% /tmp
/dev/sda2          65536        395       65141          1% /boot
/dev/sda3           0           0           0          - /home
tmpfs             115953        110      115843          1% /run/user/1000
[arzinchenko@arzinchenko ~]$ du -a -/
du: неверный ключ - «/»
По команде «du --help» можно получить дополнительную информацию.
[arzinchenko@arzinchenko ~]$ du -a ~/file.txt
176    /home/arzinchenko/file.txt
[arzinchenko@arzinchenko ~]$ du -a ~/conf.txt
8      /home/arzinchenko/conf.txt

```

Рис. 3.16: df и du

Прочитала справку команды `find`. (рис. 3.17).

```
FIND(1)                                General Commands Manual                                FIND(1)

NAME
  find - search for files in a directory hierarchy

SYNOPSIS
  find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [expression]

DESCRIPTION
  This manual page documents the GNU version of find. GNU find searches the directory tree rooted at each given starting-point by evaluating the given expression from left to right, according to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand side is false for and operations, true for or), at which point find moves on to the next file name. If no starting-point is specified, '.' is assumed.

  If you are using find in an environment where security is important (for example if you are using it to search directories that are writable by other users), you should read the 'Security Considerations' chapter of the findutils documentation, which is called Finding Files and comes with findutils. That document also includes a lot more detail and discussion than this manual page, so you may find it a more useful source of information.

OPTIONS
  The -H, -L and -P options control the treatment of symbolic links. Command-line arguments following these are taken to be names of files or directories to be examined, up to the first argument that begins with '-', or the argument '(' or '!'. That argument and any following arguments are taken to be the expression describing what is to be searched for. If no paths are given, the current directory is used. If no expression is given, the expression -print is used (but you should probably consider using -print0 instead, anyway).

  This manual page talks about 'options' within the expression list. These options control the behaviour of find but are specified immediately after the last path name. The five 'real' options -H, -L, -P, -D and -O must appear before the first path name, if at all. A double dash -- could theoretically be used to signal that any remaining arguments are not options, but this does not really work due to the way find determines the end of the following path arguments: it does that by reading until an ex-
```

Рис. 3.17: Спарвка команды `find`

Вывела имена всех директорий, имеющихя в домашнем каталоге. (рис. 3.18).


```
/home/arzinchenko/dotfiles/.git/refs/heads
/home/arzinchenko/dotfiles/.git/refs/tags
/home/arzinchenko/dotfiles/.git/refs/remotes
/home/arzinchenko/dotfiles/.git/refs/remotes/origin
/home/arzinchenko/dotfiles/.git/objects
/home/arzinchenko/dotfiles/.git/objects/pack
/home/arzinchenko/dotfiles/.git/objects/info
/home/arzinchenko/dotfiles/.git/logs
/home/arzinchenko/dotfiles/.git/logs/refs
/home/arzinchenko/dotfiles/.git/logs/refs/remotes
/home/arzinchenko/dotfiles/.git/logs/refs/remotes/origin
/home/arzinchenko/dotfiles/.git/logs/refs/heads
/home/arzinchenko/dotfiles/dot_bashrc.d
/home/arzinchenko/dotfiles/dot_config
/home/arzinchenko/dotfiles/dot_config/dunst
/home/arzinchenko/dotfiles/dot_config/fontconfig
/home/arzinchenko/dotfiles/dot_config/foot
/home/arzinchenko/dotfiles/dot_config/sway
/home/arzinchenko/dotfiles/dot_config/sway/bin
/home/arzinchenko/dotfiles/dot_config/sway/config.d
/home/arzinchenko/dotfiles/dot_config/sway/other
/home/arzinchenko/dotfiles/dot_config/sway/other/blocks
/home/arzinchenko/dotfiles/dot_config/sway/other/fuzzel
/home/arzinchenko/dotfiles/dot_config/sway/other/tessen
/home/arzinchenko/dotfiles/dot_config/sway/other/waybar
/home/arzinchenko/dotfiles/dot_config/sway/other/waybar/modules
/home/arzinchenko/dotfiles/dot_config/sway/other/wofi
/home/arzinchenko/dotfiles/dot_config/sway/scripts
/home/arzinchenko/dotfiles/dot_config/systemd
/home/arzinchenko/dotfiles/dot_config/systemd/user
/home/arzinchenko/dotfiles/dot_config/tmux
/home/arzinchenko/Anastasiia
/home/arzinchenko/reports
/home/arzinchenko/reports/monthly
/home/arzinchenko/reports/monthly/monthly
/home/arzinchenko/monthly
/home/arzinchenko/ski.places
/home/arzinchenko/ski.places/equipment
/home/arzinchenko/ski.places/plans
/home/arzinchenko/australia
/home/arzinchenko/play
/home/arzinchenko/play/games
```

Рис. 3.18: Имена всех директорий

Контрольные вопросы 1. Какие потоки ввода вывода вы знаете? В системе по умолчанию открыто три специальных потока:

stdin стандартный поток ввода (по умолчанию: клавиатура), файловый дескриптор 0;
stdout стандартный поток вывода (по умолчанию: консоль), файловый дескриптор 1;
stderr стандартный поток вывод сообщений об ошибках (по умолчанию: консоль), файл

2. Объясните разницу между операцией > и ».

Символ > используется для перенаправления вывода команды в файл. Если файл уже существует, его содержимое будет полностью перезаписано. Символ » используется для перенаправления вывода команды в файл, который не существует. Если файл уже существует, его содержимое будет добавлено к существующему.

вол » также используется для перенаправления вывода команды в файл, но с дополнением данных в конец файла, не перезаписывая существующее содержимое.

3. Что такое конвейер?

Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

4. Что такое процесс? Чем это понятие отличается от программы?

Главное отличие между программой и процессом заключается в том, что программа - это набор инструкций, который позволяет ЦПУ выполнять определенную задачу, в то время как процесс - это исполняемая программа.

5. Что такое PID и GID?

PPID - (parent process ID) идентификатор родительского процесса. Процесс может порождать и другие процессы. UID, GID - реальные идентификаторы пользователя и его группы, запустившего данный процесс.

6. Что такое задачи и какая команда позволяет ими управлять?

Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач.

7. Найдите информацию об утилитах top и htop. Каковы их функции?

Команда htop похожа на команду top по выполняемой функции: они обе показывают информацию о процессах в реальном времени, выводят данные о потреблении системных ресурсов и позволяют искать, останавливать и управлять процессами. У обеих команд есть свои преимущества. Например, в программе htop реализован очень удобный поиск по процессам, а также их фильтрация. В

команде `top` это не так удобно — нужно знать кнопку для вывода функции поиска. Зато в `top` можно разделять область окна и выводить информацию о процессах в соответствии с разными настройками. В целом `top` намного более гибкая в настройке отображения процессов.

8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.

Команда `find` - это одна из наиболее важных и часто используемых утилит системы Linux. Это команда для поиска файлов и каталогов на основе специальных условий. Ее можно использовать в различных обстоятельствах, например, для поиска файлов по разрешениям, владельцам, группам, типу, размеру и другим подобным критериям. Утилита `find` предустановлена по умолчанию во всех Linux дистрибутивах, поэтому вам не нужно будет устанавливать никаких дополнительных пакетов. Это очень важная находка для тех, кто хочет использовать командную строку наиболее эффективно. Команда `find` имеет такой синтаксис: `find [папка] [параметры] критерий шаблон [действие]` Пример: `find /etc -name "h*" -print`

9. Можно ли по контексту (содержанию) найти файл? Если да, то как?

Да, можно. Синтаксис команды: `find ~ -type f -exec grep -H 'текстДляПоиска' {} ;`

10. Как определить объем свободной памяти на жёстком диске?

С помощью команды `df -h`.

11. Как определить объем вашего домашнего каталога?

С помощью команды `du -s`.

12. Как удалить зависший процесс?

С помощью команды `kill% номер задачи`

4 Выводы

Я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

Список литературы