

# Презентация по лабораторной работе №6

Дисциплина: Операционные системы

---

Шошина Е.А.

14 марта 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Шошина Евгения Александровна
- группа: НКАбд-03-22
- студент факультета физико-математических и естественных наук
- Российский университет дружбы народов
- 1132229532@pfur.ru
- <https://EAShoshina.github.io/ru/>



## Вводная часть

---

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных.  
Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

1. Осуществите вход в систему, используя соответствующее имя пользователя.
2. Запишите в файл `file.txt` названия файлов, содержащихся в каталоге `/etc`. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
3. Выведите имена всех файлов из `file.txt`, имеющих расширение `.conf`, после чего запишите их в новый текстовый файл `conf.txt`.
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа `s`? Предложите несколько вариантов, как это сделать.
5. Выведите на экран (по странично) имена файлов из каталога `/etc`, начинающиеся с символа `h`.

6. Запустите в фоновом режиме процесс, который будет записывать в файл `~/logfile` файлы, имена которых начинаются с `log`.
7. Удалите файл `~/logfile`.
8. Запустите из консоли в фоновом режиме редактор `gedit`.
9. Определите идентификатор процесса `gedit`, используя команду `ps`, конвейер и фильтр `grep`. Как ещё можно определить идентификатор процесса?
10. Прочтите справку (`man`) команды `kill`, после чего используйте её для завершения процесса `gedit`.
11. Выполните команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man`.
12. Воспользовавшись справкой команды `find`, выведите имена всех директорий, имеющих в вашем домашнем каталоге.

## Выполнение лабораторной работы

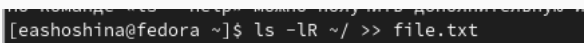
---



1. Осуществили вход в систему, используя соответствующее имя пользователя. 2. Записали в файл file.txt названия файлов, содержащихся в каталоге /etc.

```
[eashoshina@fedora ~]$ cat file.txt /etc
/etc:
итого 1448
drwxr-xr-x. 1 root root      126 ноя  5 11:51 abrt
-rw-r--r--. 1 root root      16 ноя 26 14:37 adjtime
-rw-r--r--. 1 root root    1529 июл 20  2022 aliases
drwxr-xr-x. 1 root root      70 ноя  5 12:02 alsa
drwxr-xr-x. 1 root root    1554 фев 24 22:35 alternatives
drwxr-xr-x. 1 root root      56 ноя  5 11:51 anaconda
-rw-r--r--. 1 root root     269 июл 20  2022 anthy-unicode.conf
-rw-r--r--. 1 root root     769 мая 23  2022 appstream.conf
-rw-r--r--. 1 root root      55 июл 20  2022 asound.conf
drwxr-x---. 1 root root     108 ноя 26 14:44 audit
drwxr-xr-x. 1 root root     232 ноя 26 14:37 authselect
drwxr-xr-x. 1 root root      66 ноя  5 11:38 avahi
drwxr-xr-x. 1 root root     144 ноя  5 11:51 bash_completion.d
-rw-r--r--. 1 root root    2638 июл 20  2022 bashrc
-rw-r--r--. 1 root root     535 авг  8  2022 bindresvport.blacklist
drwxr-xr-x. 1 root root       0 окт 24 22:50 binfmt.d
drwxr-xr-x. 1 root root      18 ноя  5 11:40 bluetooth
-rw-r-----. 1 root brlapi    33 ноя  5 11:47 brlapi.key
drwxr-xr-x. 1 root root      76 ноя  5 11:47 brltty
-rw-r--r--. 1 root root   29842 авг  2  2022 brltty.conf
```

Дописали в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.



```
no команде «ls -lR ~/» можно получить дополнительную и.  
[eashoshina@fedora ~]$ ls -lR ~/ >> file.txt
```

Рис. 2: Запись названий файлов из домашнего каталога /etc в файл file.txt

### 3. Вывели имена всех файлов из file.txt, имеющих расширение .conf.

```
[eashoshina@fedora ~]$ grep .conf file.txt
-rw-r--r--. 1 root root      269 июл 20  2022 anthy-unicode.conf
-rw-r--r--. 1 root root      769 мая 23  2022 appstream.conf
-rw-r--r--. 1 root root       55 июл 20  2022 asound.conf
-rw-r--r--. 1 root root    29842 авг  2  2022 brltty.conf
drwxr-xr-x. 1 root root       0 июл 21  2022 chkconfig.d
-rw-r--r--. 1 root root    1371 авг 29  2022 chrony.conf
drwxr-xr-x. 1 root root      18 ноя  5 11:38 dconf
-rw-r--r--. 1 root root    1280 июл 21  2022 dley-na-renderer-service.conf
-rw-r--r--. 1 root root    1174 июл 21  2022 dley-na-server-service.conf
-rw-r--r--. 1 root root   28442 сен 29 17:34 dnsmasq.conf
-rw-r--r--. 1 root root     117 сен 14  2022 dracut.conf
drwxr-xr-x. 1 root root       0 сен 14  2022 dracut.conf.d
-rw-r--r--. 1 root root      20 фев 24  2022 fprintd.conf
-rw-r--r--. 1 root root      38 июл 21  2022 fuse.conf
-rw-r--r--. 1 root root       9 июл 20  2022 host.conf
-rw-r--r--. 1 root root    5799 сен 27 17:52 idmapd.conf
-rw-r--r--. 1 root root    8892 ноя  5 11:47 kdump.conf
-rw-r--r--. 1 root root     880 июл 21  2022 krb5.conf
drwxr-xr-x. 1 root root     106 ноя  5 12:02 krb5.conf.d
-rw-r--r--. 1 root root      28 окт  7 15:09 ld.so.conf
drwxr-xr-x. 1 root root     116 фев 24 14:15 ld.so.conf.d
-rw-r-----. 1 root root     191 авг 30  2022 libaudit.conf
-rw-r--r--. 1 root root    2393 сен 23 20:25 libuser.conf
-rw-rw-r--. 1 root root      19 ноя 26 14:37 locale.conf
-rw-r--r--. 1 root root     493 апр 16  2021 logrotate.conf
-rw-r--r--. 1 root root    5122 авг  3  2022 makedumpfile.conf.sample
-rw-r--r--. 1 root root    5235 июл 22  2022 man_db.conf
-rw-r--r--. 1 root root     782 июл 21  2022 mke2fs.conf
-rw-r--r--. 1 root root    2620 сен 20 10:05 mtools.conf
```

Записали их в новый текстовой файл conf.txt.

```
[eashoshina@fedora ~]$ grep '\.conf' file.txt > conf.txt
```

Рис. 4: Запись этих имен в файл conf.txt

#### 4. Определили, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с?

```
[eashoshina@fedora ~]$ grep '\.conf' file.txt > conf.txt
[eashoshina@fedora ~]$ find ~ -name "c*" -print
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/crashes
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/compatibility.ini
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/cookies.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/cert9.db
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++habr.com/cache
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++habr.com/cache/caches.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++products.groupdocs.app/cache
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++products.groupdocs.app/cache/caches.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++www.youtube.com/cache
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++www.youtube.com/cache/caches.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++www.youtube.com/cache/caches.sqlite-wal
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++www.youtube.com/cache/context_open.marker
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++wiki.dieg.info/cache
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/storage/default/https+++wiki.dieg.info/cache/caches.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/content-prefs.sqlite
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/containers.json
/home/eashoshina/.mozilla/firefox/0qlthcn4.default-release/cookies.sqlite-wal
/home/eashoshina/.cache/mesa_shader_cache/ca
/home/eashoshina/.cache/mesa_shader_cache/9b/c94d7e5cc95774aa0dc3d73572288cb14c4ded
/home/eashoshina/.cache/mesa_shader_cache/65/c8ed50e21153def915daff3af21de421c68a51
/home/eashoshina/.cache/mesa_shader_cache/d6/c3dd2b252db507445983e6b5da4fcf9c463d11
/home/eashoshina/.cache/mesa_shader_cache/fb/c3cc2f8bd754e8f575c75ba2298af45e356302
/home/eashoshina/.cache/mesa_shader_cache/cf
/home/eashoshina/.cache/mesa_shader_cache/91/c4b1ee465cb7a9cae9a6913326b6baa3990064
/home/eashoshina/.cache/mesa_shader_cache/c6
/home/eashoshina/.cache/mesa_shader_cache/fc/ca1f562286344143ad5c9dcea4d77994e4ce8
```

4. Определили, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с?

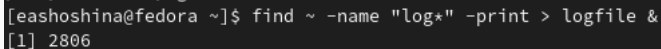
```
[eashoshina@fedora ~]$ ls -l | grep c*  
-rw-r--r--. 1 eashoshina eashoshina 41053 мар 16 20:07 conf.txt
```

Рис. 6: Команда grep

5. Вывели на экран (пос транично) имена файлов из каталога /etc, начинающиеся с символа h.

```
[eashoshina@fedora ~]$ find /etc -name "h*" -print
find: '/etc/audit': Отказано в доступе
/etc/avahi/hosts
/etc/brltty/Contraction/ha.ctb
/etc/brltty/Input/bm/horizontal.kti
/etc/brltty/Input/hd
/etc/brltty/Input/hm
/etc/brltty/Input/ht
/etc/brltty/Input/hw
/etc/brltty/Text/he.ttb
/etc/brltty/Text/hi.ttb
/etc/brltty/Text/hr.ttb
/etc/brltty/Text/hu.ttb
/etc/brltty/Text/hy.ttb
/etc/containers/oci/hooks.d
find: '/etc/cups/ssl': Отказано в доступе
find: '/etc/dhcp': Отказано в доступе
find: '/etc/firewalld': Отказано в доступе
find: '/etc/grub.d': Отказано в доступе
/etc/hp
/etc/hp/hplip.conf
/etc/httpd
/etc/httpd/conf/httpd.conf
```

6. Запустили в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.

A terminal window with a dark background. The prompt is [eashoshina@fedora ~]. The command entered is find ~ -name "log\*" -print > logfile &. The output is [1] 2806.

```
[eashoshina@fedora ~]$ find ~ -name "log*" -print > logfile &  
[1] 2806
```

Рис. 8: Запуск процесса в фоновом режиме



## 7. Удалили файл ~/logfile.

```
[eashoshina@fedora ~]$ rm logfile  
[1]+  Завершён          find ~ -name "log*" -print > logfile
```

Рис. 9: Удаление файла ~/logfile

## 8. Запустили из консоли в фоновом режиме редактор gedit.

A terminal window with a black background and white text. The prompt is [eashoshina@fedora ~]\$. The command gedit & has been entered and executed. The output is [1] 2824, indicating the process ID of the background gedit process.

```
[eashoshina@fedora ~]$ gedit &  
[1] 2824
```

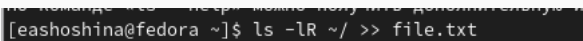
Рис. 10: Запуск из консоли в фоновом режиме редактора gedit

9. Определили идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?

```
[eashoshina@fedora ~]$ ps aux | grep gedit
eashosh+   2882  2.2  1.8 792244 73992 pts/0    Sl   20:16   0:00 gedit
eashosh+   2933  0.0  0.0 222036  2372 pts/0    S+   20:17   0:00 grep --color=auto
gedit
[eashoshina@fedora ~]$ pgrep gedit
2882
[eashoshina@fedora ~]$ ps aux | grep gedit | grep -v grep
eashosh+   2882  1.3  1.8 792244 73992 pts/0    Sl   20:16   0:00 gedit
```

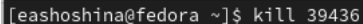
Рис. 11: Определение идентификатора процесса gedit

10. Прочитали справку (man) команды kill, после чего использовали её для завершения процесса gedit.



```
[eashoshina@fedora ~]$ ls -lR ~/ >> file.txt
```

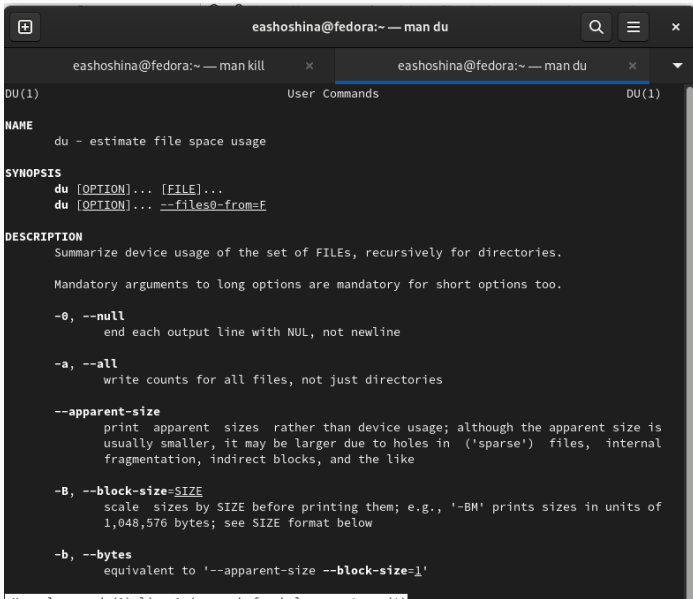
Рис. 12: Справка man команды kill



```
[eashoshina@fedora ~]$ kill 39436
```

Рис. 13: Команда kill

## 11. Выполнили команду man du



```
eashoshina@fedora:~ — man du
eashoshina@fedora:~ — man kill
eashoshina@fedora:~ — man du
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

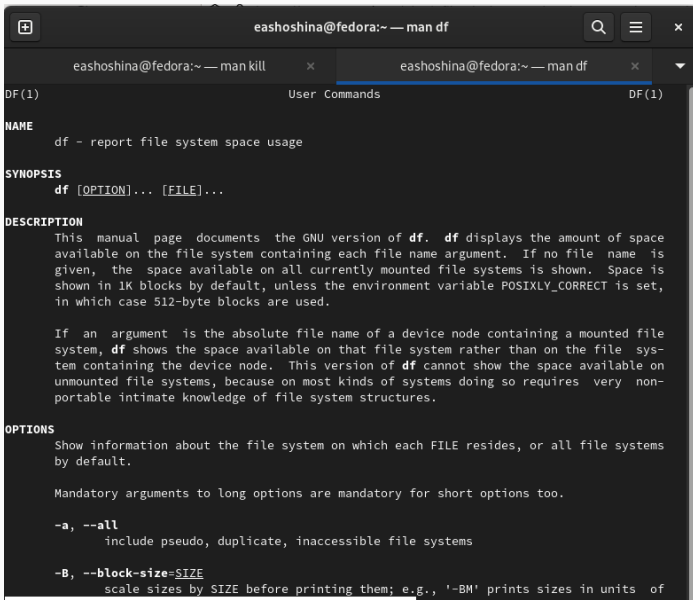
-a, --all
    write counts for all files, not just directories

--apparent-size
    print apparent sizes rather than device usage; although the apparent 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'
```

## 11. Выполнили команду man df



```
eashoshina@fedora:~ — man df
eashoshina@fedora:~ — man kill x eashoshina@fedora:~ — man df x
DF(1) User Commands DF(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 sys-
    tem 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 non-
    portable 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
```

## 11. Выполнили команду df- vi

```
[eashoshina@fedora ~]$ df -vi
```

Файловая система	Инодов	ИИспользовано	ИСвободно	ИИспользовано%	Смонтировано в
devtmpfs	1048576	461	1048115	1%	/dev
tmpfs	501515	10	501505	1%	/dev/shm
tmpfs	819200	890	818310	1%	/run
/dev/sda3	0	0	0	-	/
tmpfs	1048576	46	1048530	1%	/tmp
/dev/sda2	65536	387	65149	1%	/boot
/dev/sda3	0	0	0	-	/home
tmpfs	100303	152	100151	1%	/run/user/1000

Рис. 16: Команда df- vi

## 11. Выполнили команду du -a

```
eashoshina@fedora:~ — man kill × eashoshina@fedora:~ ×
8      ./local/lib/python3.11/site-packages/pandas/tests/__pycache__/test_register_accessor.cpython-311.pyc
36     ./local/lib/python3.11/site-packages/pandas/tests/__pycache__/test_sorting.cpython-311.pyc
24     ./local/lib/python3.11/site-packages/pandas/tests/__pycache__/test_take.cpython-311.pyc
384    ./local/lib/python3.11/site-packages/pandas/tests/__pycache__/
35284  ./local/lib/python3.11/site-packages/pandas/tests
76     ./local/lib/python3.11/site-packages/pandas/io/sql.py
4      ./local/lib/python3.11/site-packages/pandas/io/date_converters.py
12     ./local/lib/python3.11/site-packages/pandas/io/gbq.py
168    ./local/lib/python3.11/site-packages/pandas/io/pytables.py
36     ./local/lib/python3.11/site-packages/pandas/io/xml.py
20     ./local/lib/python3.11/site-packages/pandas/io/parquet.py
4      ./local/lib/python3.11/site-packages/pandas/io/spss.py
4      ./local/lib/python3.11/site-packages/pandas/io/api.py
4      ./local/lib/python3.11/site-packages/pandas/io/feather_format.py
8      ./local/lib/python3.11/site-packages/pandas/io/clipboards.py
8      ./local/lib/python3.11/site-packages/pandas/io/orc.py
40     ./local/lib/python3.11/site-packages/pandas/io/html.py
40     ./local/lib/python3.11/site-packages/pandas/io/common.py
132    ./local/lib/python3.11/site-packages/pandas/io/stata.py
8      ./local/lib/python3.11/site-packages/pandas/io/pickle.py
4      ./local/lib/python3.11/site-packages/pandas/io/__init__.py
36     ./local/lib/python3.11/site-packages/pandas/io/formats/info.py
8      ./local/lib/python3.11/site-packages/pandas/io/formats/_color_data.py
28     ./local/lib/python3.11/site-packages/pandas/io/formats/latex.py
72     ./local/lib/python3.11/site-packages/pandas/io/formats/format.py
4      ./local/lib/python3.11/site-packages/pandas/io/formats/console.py
16     ./local/lib/python3.11/site-packages/pandas/io/formats/xml.py
156    ./local/lib/python3.11/site-packages/pandas/io/formats/style.py
```



12. Воспользовались справкой команды `find`, вывели имена всех директорий, имеющих в вашем домашнем каталоге.

```
eashoshina@fedora:~ — man kill × eashoshina@fedora:~ — man find ×
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
```

12. Воспользовались справкой команды `find`, вывели имена всех директорий, имеющих в вашем домашнем каталоге.

```
[eashoshina@fedora ~]$ find -type d
.
./.mozilla
./.mozilla/extensions
./.mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
./.mozilla/plugins
./.mozilla/firefox
./.mozilla/firefox/Crash Reports
./.mozilla/firefox/Crash Reports/events
./.mozilla/firefox/Pending Pings
./.mozilla/firefox/0qlthcn4.default-release
./.mozilla/firefox/0qlthcn4.default-release/minidumps
./.mozilla/firefox/0qlthcn4.default-release/crashes
./.mozilla/firefox/0qlthcn4.default-release/crashes/events
./.mozilla/firefox/0qlthcn4.default-release/security_state
./.mozilla/firefox/0qlthcn4.default-release/storage
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb/387011
2724rsegmnoittet-es.files
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb/387011
2724rsegmnoittet-es.files/journals
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb/356128
8849sdhlie.files
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb/145131
8868ntouromlalnody--epcr.files
./.mozilla/firefox/0qlthcn4.default-release/storage/permanent/chrome/idb/165711
```

## Выводы

---

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

## 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), файловый дескриптор 2. Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода `stdout`. Например, команда `ls` выводит в стандартный поток вывода (консоль) список файлов в текущей директории. Потоки вывода и ввода можно перенаправлять на другие файлы или устройства. Проще всего это делается с помощью символов `>`, `»`, `<`, `«`.

## 2. Объясните разницу между операцией > и ».

- В Bash знак больше > обозначает перенаправление стандартного потока вывода. В данном случае в файл. То есть cat по-умолчанию выводит данные на экран, но поскольку они были перенаправлены, то данные на экран выводиться не будут. На экране видны только вводимые строки, выводимые оказываются в файле. Два знака больше » – это тоже перенаправление вывода, но такое, когда данные добавляются в конец объекта (в данном случае файла), если он существует. Используем мы только один знак больше, файл был бы перезаписан.

### 3. Что такое конвейер?

- Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий: команда 1 | команда 2 # означает, что вывод команды 1 передаётся на ввод команде 2 Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например: `ls -la | sort > sorting_list` вывод команды `ls -la` передаётся команде сортировки `sort`, которая пишет результат в файл `sorting_list`.



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

Программа и процесс актуальны, но отличаются. Программа - это всего лишь сценарий, хранящийся на диске или, по-видимому, предыдущий этап процесса. Наоборот, процесс является событием программы в процессе выполнения. Команда `ps` используется для получения информации о процессах. Для получения информации о процессах, управляемых вами и запущенных (работающих или остановленных) на вашем терминале, используйте опцию `aux`.

## 5. Что такое PID и GID?

- Идентификатор процесса (PID). Каждому новому процессу ядро присваивает уникальный идентификационный номер. В любой момент времени идентификатор процесса является уникальным, хотя после завершения процесса он может использоваться снова для другого процесса. Некоторые идентификаторы зарезервированы системой для особых процессов. Так, процесс с идентификатором 1 - это процесс инициализации `init`, являющийся предком всех других процессов в системе.
- Идентификатор группы GID и эффективный идентификатор группы (EGID) GID - это идентификационный номер группы данного процесса. EGID связан с GID также, как EUID с UID.

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

Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать знак амперсанда &.

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

- `top` — самая простая и самая распространённая утилита из этого списка. Показывает примерно то же, что утилита `vmstat`, плюс рейтинг процессов по потреблению памяти или процессора. Совсем ничего не знает про загрузку сети или дисков. Позволяет минимальный набор операций с процессом: `renice`, `kill`.
- `htop` не собирает статистику и просто показывает текущее состояние. Второе яркое отличие — нортоноподобная панелька с подсказками кнопок снизу и возможность «навигации» по списку процессов.

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

- `find` : Для поиска файлов из командной строки вы можете использовать команду `find`. У этой команды следующий синтаксис: `find path criteria action` `path` - Секция для указания директории поиска. Если ничего не указано поиск идет по текущей директории. `criteria` - Опции поиска. `action` - Опции, которые влияют на состояние поиска или контролируют его, например, `-print`

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

- Для поиска файла по содержимому проще всего воспользоваться командой `grep` (вместо `find`).

## 10. Как определить объем свободной памяти на жёстком диске?

-Команда `df` — сокращенное «disk-free», показывает доступное и используемое дисковое пространство в системе Linux.

## 11. Как определить объем вашего домашнего каталога?

- Для просмотра размеров папок на диске используется команда `du`. Если просто ввести команду без каких либо аргументов, то она рекурсивно проскандирует вашу текущую директорию и выведет размеры всех файлов в ней. Обычно для `du` указывают путь до папки, которую вы хотите проанализировать.



## 12. Как удалить зависший процесс?

- Когда известен PID процесса, мы можем убить его командой `kill`.
- Утилита `kill` - это оболочка для `kill`, она ведет себя точно так же, и имеет тот же синтаксис, только в качестве идентификатора процесса ей нужно передать его имя.
- Команда `killall` в Linux предназначена для «убийства» всех процессов, имеющих одно и то же имя.