

# Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Лабораторная работа №6

---

Толстых М. А.

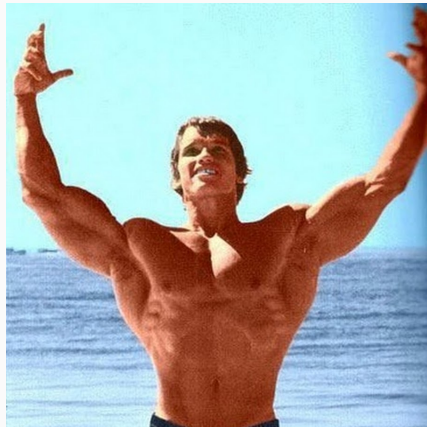
17 марта 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Толстых Максим Алексеевич
- студент 1 курса, группа НММбд-03-22
- Российский университет дружбы народов



## Вводная часть

---

- Инструменты поиска файлов и фильтрации текстовых данных
- Файлы и текстовые данные

- 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.

- Ознакомиться и разобрать на практике основные инструменты поиска файлов и фильтрации текстовых данных.
- Выполнить упражнения.
- Ответить на контрольные вопросы.

## Выполнение лабораторной работы №6

---



# Команды поиска и фильтрации

```
[matolstikh@fedora ~]$ find /etc > file.txt
```

```
[matolstikh@fedora ~]$ find ~ >> file.txt
```

```
[matolstikh@fedora ~]$ grep ".conf$" file.txt  
/etc/extlinux.conf  
/etc/opensc-x86_64.conf  
/etc/NetworkManager/NetworkManager.conf  
/etc/PackageKit/CommandNotFound.conf  
/etc/PackageKit/PackageKit.conf  
/etc/PackageKit/Viewer.conf  
/etc/UPower/UPower.conf  
/etc/X11/xinit/xinput.d/lbus.conf  
/etc/X11/xorg.conf.d/00-keyboard.conf  
/etc/abrt/plugins/CCpp.conf  
/etc/abrt/plugins/java.conf  
/etc/abrt/plugins/oops.conf  
/etc/abrt/plugins/python3.conf  
/etc/abrt/plugins/vmcore.conf  
/etc/abrt/plugins/xorg.conf
```

# Запуск команд в фоновом режиме

```
[matolstikh@fedora ~]$ ls -lR | grep log > logfile &
```

```
[matolstikh@fedora ~]$ rm logfile  
[]* 3ae9p8W      ls --color=auto -lR | grep --color=auto log > logfile
```

```
[matolstikh@fedora ~]$ gedit &  
[] 2939
```

# Определение идентификатора процесса

```
[matolstikh@fedora ~]$ ps aux | grep gedit
matolst 2981 0.0 1.5 783624 74812 pts/0    Sl   15:47   0:00 gedit
matolst 3041 0.0 0.0 222192 2124 pts/0    S+   15:50   0:00 grep --color=auto gedit
```

```
[matolstikh@fedora ~]$ pgrep gedit
2981
```

```
[matolstikh@fedora ~]$ pidof gedit
2981
```

# Комманда kill

```
[matolstikh@fedora ~]$ man kill
```

```
matolstikh@fedora:~$ man kill
kill(1) If signal is 0, then no actual signal is sent, but error checking is still performed.
SYNOPSIS
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:
  -p pid where pid is larger than 0. The process with PID pid is signaled.
  -g pid All processes in the current process group are signaled.
  -P All processes with a PID larger than 1 are signaled.
  -P pid where pid is larger than 1. All processes in process group pid are signaled. When
  as argument of the form '-P pid' is given, and it is meant to denote a process
  group, either a signal must be specified first, or the argument must be
  preceded by a '--' option, otherwise it will be taken as the signal to send.
```

```
bash: kill: %2981: not found
[matolstikh@fedora ~]$ kill 2981
```

# Команды df, du

```
[matolstikh@fedora ~]$ man df
```

```
matolstikh@fedora:~$ man df

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
    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
    print sizes in powers of 1000 (e.g., 1.1G)

-i, --inodes
    list inode information instead of block usage

-k      like --block-size=K

Manual page df(1) line 25 (press h for help or q to quit)
```

```
[matolstikh@fedora ~]$ man du
```

# Команды df, du

```
matolstikh@fedora:~$ man du
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'

-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 --summa-
Manual page du(1), line 17 (press h for help or q to quit)
```

```
matolstikh@fedora ~$ df
Байт/сек Система 1K-блоков Использовано Доступно Использованно Ожидается в
devtmpfs 4096 0 4096 0% /dev
tmpfs 2393436 0 2393436 0% /dev/shm
tmpfs 907276 2408 905868 1% /run
/dev/sda2 82836480 13896796 68939784 17% /
tmpfs 2393436 56 2393380 1% /tmp
/dev/sda2 82836480 13896796 68939784 17% /home
/dev/sda1 900780 230108 677666 26% /boot
Hwem_hayase 123901848 110673392 132268456 90% /media/sf_
p4q 123901848 110673392 132268456 90% /media/sf_
tmpfs 476684 212 476472 1% /run/user/1090
```

```
matolstikh@fedora ~$ du
0 ./mozilla/extensions/fe0803077-c20a-404f-9b0e-136329e07364)
0 ./mozilla/extensions
0 ./mozilla/plugins
0 ./mozilla/firefox/Crash Reports/events
4 ./mozilla/firefox/Crash Reports
0 ./mozilla/firefox/Pending Pings
0 ./mozilla/firefox/t5j0aggy.default-release/minidumps
0 ./mozilla/firefox/t5j0aggy.default-release/crashes/events
4 ./mozilla/firefox/t5j0aggy.default-release/crashes
2690 ./mozilla/firefox/t5j0aggy.default-release/security_state
0 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/3870112724rswgn
o1ttet-es-files/journals
792 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/3870112724rswgn
o1ttet-es-files
0 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/356128849sdhli
-files
0 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/245131868dtour
olalndry--pcr-files
0 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/1657114595Amcat
svt5ty-files
0 ./mozilla/firefox/t5j0aggy.default-release/storage/permanent/chrome/sdb/2623118077thour
```

# Команда find

```
[matolstikh@fedora ~]$ man find
```

```
matolstikh@fedora:~$ man find

-P      Never follow symbolic links. This is the default behaviour. When find examines
or prints information about files, and the file is a symbolic link, the information
used shall be taken from the properties of the symbolic link itself.

-l      Follow symbolic links. When find examines or prints information about files,
the information used shall be taken from the properties of the file to which the
link points, not from the link itself (unless it is a broken symbolic link or
find is unable to examine the file to which the link points). Use of this
option implies -noleaf. If you later use the -P option, -noleaf will still be in
effect. If -l is in effect and find discovers a symbolic link to a subdirectory
during its search, the subdirectory pointed to by the symbolic link will be
searched.

-t      When the -t option is in effect, the -type predicate will always match against
the type of the file that a symbolic link points to rather than the link itself
(unless the symbolic link is broken). Actions that can cause symbolic links to
become broken while find is executing (for example delete) can give rise to
confusing behaviour. Using -t causes the -lname and -ilname predicates always
to return false.

-H      Do not follow symbolic links, except while processing the command line
arguments. When find examines or prints information about files, the information
Retain page find.1: Line 46 (press h for help or q to quit)
```

```
[matolstikh@fedora ~]$ find -type d -depth -print
find: warning: you have specified the global option -depth after the argument -type, but global
options are not positional, i.e., -depth affects tests specified before it as well as those spec
ified after it. Please specify global options before other arguments.
/home/matolstikh/.mozilla/extensions/ec8030f7-c20a-464f-80be-11a39e973841
/home/matolstikh/.mozilla/extensions
/home/matolstikh/.mozilla/plugins
/home/matolstikh/.mozilla/firefox/Crash Reports/events
/home/matolstikh/.mozilla/firefox/Crash Reports
/home/matolstikh/.mozilla/firefox/Pending Pings
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/minidumps
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/crashes/events
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/crashes
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/security_State
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/storage/permanent/chrome/ldb/38701127
drusegnotttet-es_files/journals
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/storage/permanent/chrome/ldb/38701127
drusegnotttet-es_files
/home/matolstikh/.mozilla/firefox/t9jbsqgy.default-release/storage/permanent/chrome/ldb/35612808
tsdfile_files
```

1. Какие потоки ввода вывода вы знаете?

- `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.

2. Объясните разницу между операцией `>` и `> >`. `>` - создаётся файл и в него записываются данные; `> >` - файл открывается в режиме добавления.

3. Что такое конвейер?

Конвейер (англ. *pipeline*) в терминологии операционных систем семейства Unix — некоторое множество процессов, для которых выполнено следующее перенаправление ввода-вывода: то, что выводит на поток стандартного вывода предыдущий процесс, попадает в поток стандартного ввода следующего процесса.

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

- `Top` : Программа позволяет интерактивно просматривать список запущенных процессов Linux.
- `Htop`: Это ещё более мощная утилита для просмотра запущенных процессов в Linux. Пользоваться ею намного удобнее. Здесь поддерживаются не только горячие клавиши, но и управление мышью. Она выводит всё в цвете, поэтому смотреть на данные намного приятнее.

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

Команда `find` используется для поиска и отображения на экран имён файлов, соответствующих заданной строке символов.

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

В Linux всё это делается с помощью одной весьма простой, но в то же время мощной утилиты `grep`. С её помощью можно искать не только строки в файлах, но и фильтровать вывод команд.

10. Как определить объем свободной памяти на жёстком диске?(`df`)



## Результаты

---

В ходе выполнения лабораторной работы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.