

# Лабораторная работа № 8. Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Отчёт

---

Сергеев Д. О.

05 апреля 2025

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Сергеев Даниил Олегович
- Студент
- Направление: Прикладная информатика
- Российский университет дружбы народов
- 1132246837@pfur.ru

## Цель работы

---

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.

## Задание

---

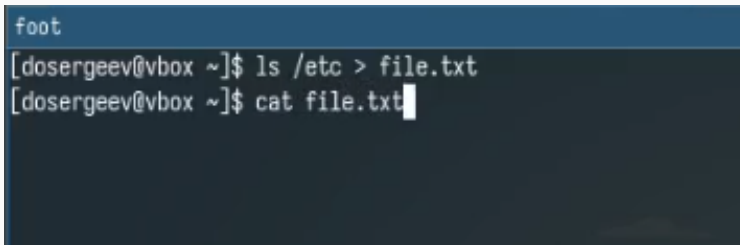
1. Управление потоком вывода и его запись в файлы
2. Выполнить задания, использующие фильтр `grep` и команду поиска `find`
3. Выполнить задания, связанные с фоновыми процессами.
4. Использовать команды `df` и `du`.
5. Поработать с командой `find` для поиска директорий

## Ход выполнения лабораторной работы

---



Запишем в файл file.txt названия файлов из каталога /etc. Для этого используем команду ls и операцию перенаправления вывода (>).

A terminal window with a dark background and a blue header bar containing the text 'foot'. The terminal shows two commands being executed. The first command is '[dosergeev@vbox ~]\$ ls /etc > file.txt' and the second command is '[dosergeev@vbox ~]\$ cat file.txt'. A white cursor is visible at the end of the second command.

```
foot
[dosergeev@vbox ~]$ ls /etc > file.txt
[dosergeev@vbox ~]$ cat file.txt
```

Рис. 1: Запись файлов в file.txt

## Выполнение примеров из описания лабораторной работы

Теперь дополнительно запишем в этот файл названия файлов, содержащихся в домашнем каталоге. Для этого используем перенаправление вывода с функцией добавления в конец файла (»).

```
X11
xattr.conf
xdg
x12tpd
xml
yum.repos.d
[dosergeev@vbox ~]$ ls ~/ >> file.txt
[dosergeev@vbox ~]$ cat file.txt
```

Рис. 2: Добавление файлов в конец file.txt

```
Видео  
Документы  
Загрузки  
Изображения  
Музыка  
Общедоступные  
Рабочий стол  
Шаблоны  
[doseergeev@vbox ~]$
```



Рис. 3: Конец вывода file.txt

## Выполнение примеров из описания лабораторной работы

Выведем имена всех файлов из file.txt, имеющих расширение .conf и запишем их в новый файл conf.txt. Для этого пропишем команду `cat file.txt | grep "\.conf$" > conf.txt`. Символ `$` нам нужен для поиска в конце имен, а символ `.` для указания того, что `.` - не системный символ.

```
foot
[dosergeev@vbox ~]$ cat file.txt | grep "\.conf$"
asound.conf
chrony.conf
dnsmasq.conf
dracut.conf
fprintd.conf
fuse.conf
host.conf
idmapd.conf
ipsec.conf
kdump.conf
```

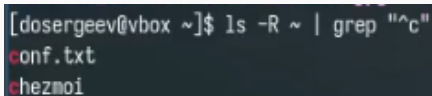
Рис. 4: Файлы формата .conf

```
[dosergeev@vbox ~]$ cat file.txt | grep "\.conf$" > conf.txt  
[dosergeev@vbox ~]$ cat conf.txt  
asound.conf  
chrony.conf  
dnsmasq.conf  
dracut.conf
```

Рис. 5: Запись результата в conf.txt

Определим, какие файлы в домашнем каталоге имеют имена, начинающиеся с символа с. Для этого используем два варианта:

1. `ls -R ~ | grep "^с"`
2. `find ~ -name "с*" -print`



```
[dosergeev@vbox ~]$ ls -R ~ | grep "^с"  
сnf.txt  
сhezmoi
```

Рис. 6: Первый вариант вывода

```
[dosergeev@vbox ~]$ find ~ -name "c*" -print  
/home/dosergeev/.mozilla/firefox/ximan1b.default-release/crashes  
/home/dosergeev/.mozilla/firefox/ximan1b.default-release/compatibility.ini  
/home/dosergeev/.mozilla/firefox/ximan1b.default-release/cookies.sqlite
```

Рис. 7: Второй вариант вывода

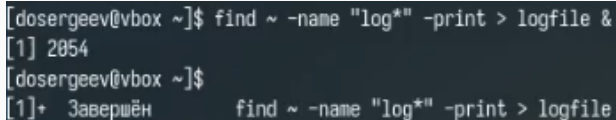
## Выполнение примеров из описания лабораторной работы

Выведем на экран по странично имена файлов из /etc, начинающиеся с символа h. Для вывода по странично используем конвейер и команду more.

```
[dosergeev@vbox ~]$ sudo find /etc -name "h*" | more
[sudo] пароль для dosergeev:
/etc/avahi/hosts
/etc/firewalld/helpers
/etc/libibverbs.d/hfi1verbs.driver
/etc/libibverbs.d/hns.driver
/etc/nvme/hostnqn
/etc/nvme/hostid
/etc/udev/hwdb.d
/etc/udev/hwdb.bin
/etc/host.conf
/etc/hosts
/etc/hostname
/etc/mercurial/hgrc.d
```



Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. Удалим созданный файл.



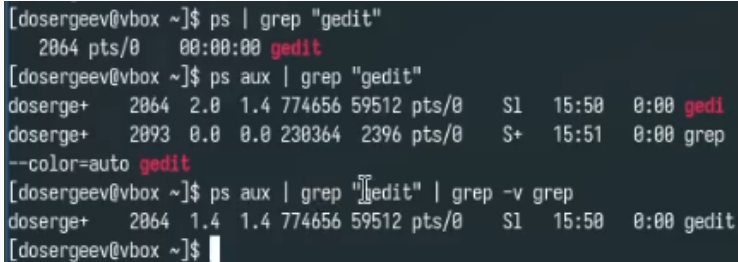
```
[dosergeev@vbox ~]$ find ~ -name "log*" -print > logfile &  
[1] 2854  
[dosergeev@vbox ~]$  
[1]+  Завершён      find ~ -name "log*" -print > logfile
```

Рис. 9: Фоновый процесс для записи log\*.

## Выполнение примеров из описания лабораторной работы

Запустим из консоли в фоновом режиме редактор gedit с помощью команды `gedit &`.  
Определим его идентификатор, используя `ps`, конвейер и фильтр `grep`.

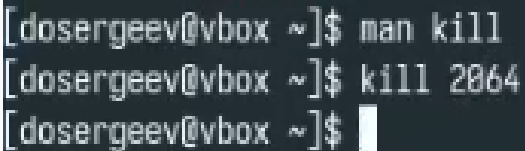
1. `ps | grep "gedit"`
2. `ps aux | grep "gedit" | grep -v grep`



```
[dosergeev@vbox ~]$ ps | grep "gedit"
  2064 pts/0    00:00:00 gedit
[dosergeev@vbox ~]$ ps aux | grep "gedit"
doserge+  2064  2.0  1.4 774656 59512 pts/0    S1   15:50   0:00 gedi
doserge+  2093  0.0  0.0 230364  2396 pts/0    S+   15:51   0:00 grep
--color=auto gedit
[dosergeev@vbox ~]$ ps aux | grep "gedit" | grep -v grep
doserge+  2064  1.4  1.4 774656 59512 pts/0    S1   15:50   0:00 gedit
[dosergeev@vbox ~]$
```

Рис. 10: Поиск процесса gedit

Завершим процесс командой kill, предварительно прочитав её описание

A terminal window with a dark background and light-colored text. It shows three lines of command execution. The first line is '[dosergeev@vbox ~]\$ man kill', the second is '[dosergeev@vbox ~]\$ kill 2864', and the third is '[dosergeev@vbox ~]\$' followed by a cursor. The prompt character is a light blue square.

```
[dosergeev@vbox ~]$ man kill  
[dosergeev@vbox ~]$ kill 2864  
[dosergeev@vbox ~]$
```

Рис. 11: Завершение gedit.

```
foot
[dosergeev@vbox ~]$ ps aux | grep "gedit" | grep -v grep
[1]+  Завершено      gedit
[dosergeev@vbox ~]$
```

Рис. 12: Проверка процесса.

## Выполнение примеров из описания лабораторной работы

Выполним команды `df` и `du`, предварительно получив более подробную информацию из `man`. Используем `df` и `du` с ключем `-h` для вывода размера файлов и файловых систем в понятном для человека формате. К `du` добавим ключ `-a` для вывода размера файлов включительно.

```
[dosergeev@vbox ~]$ df -h
```

Файловая система	Размер	Использовано	Дост	Использовано%	Смонтировано в
/dev/sda3	79G	9,9G	68G	13%	/
devtmpfs	4,0M	0	4,0M	0%	/dev
tmpfs	1,9G	3,4M	1,9G	1%	/dev/shm
efivarfs	256K	221K	31K	88%	/sys/firmware/efi/efivars
tmpfs	778M	1,3M	777M	1%	/run
tmpfs	1,0M	0	1,0M	0%	/run/credentials/systemd-journald.service
tmpfs	1,0M	0	1,0M	0%	/run/credentials/systemd-network-generator.service
tmpfs	1,0M	0	1,0M	0%	/run/credentials/systemd-udev-load-credentials.service

Рис. 13: Команда `df`.

## Выполнение примеров из описания лабораторной работы

```
8,0K  ./ski.plases/equipment/equiplist
0     ./ski.plases/equipment/equiplist2
8,0K  ./ski.plases/equipment
0     ./ski.plases/plans
8,0K  ./ski.plases
0     ./abc1
0     ./lab07/my_os
0     ./lab07/feathers
0     ./lab07/australia
0     ./lab07/play
0     ./lab07
0     ./play/file.old
0     ./play/games/file.old
0     ./play/games
0     ./play
4,0K  ./vboxclient-clipboard-tty2-service.pid
4,0K  ./file.txt
4,0K  ./conf.txt
24K   ./bash_history
1,1G  .
[dosergeev@vbox ~]$
```

Рис. 14: Вывод команды du.

## Выполнение примеров из описания лабораторной работы

Воспользовавшись справкой команды `find`, выведем имена всех директорий, имеющихя в домашнем каталоге. Для этого используем `find ~ -type d -print`, где `-type` - ключ для выбора типа искомого объекта.

```
/home/dosergeev/reports/monthly
/home/dosergeev/monthly
/home/dosergeev/ski.places
/home/dosergeev/ski.places/equipment
/home/dosergeev/ski.places/plans
/home/dosergeev/lab07
/home/dosergeev/lab07/australia
/home/dosergeev/lab07/play
/home/dosergeev/play
/home/dosergeev/play/games
[dosergeev@vbox ~]$
```

Рис. 15: Вывод команды `find`.

1.

- `stdin` — поток ввода, по умолчанию: клавиатура, имеет сигнал 0;
- `stdout` — поток вывода, по умолчанию: консоль, имеет сигнал 1;
- `stderr` — поток вывода сообщений об ошибках, по умолчанию: консоль, имеет сигнал 2;

2.

- `'>'` — перенаправление вывода с перезаписью файла.
- `'>>'` — перенаправление вывода с дописыванием в конец файла.



3. Конвейер (|) – это символ для передачи вывода одной команды на вход другой.
4. Программа – исполняемый файл на диске. Процесс – экземпляр запущенной программы в памяти, имеющий собственный номер, свои ресурсы и состояние.
5. PID (Process ID) – уникальный идентификатор процесса. GID (Group ID) – идентификатор группы процессов.
6. Задачи - процессы, запущенные в текущей сессии терминала. kill %номер задачи - позволяет завершить процесс по номеру.
7. top – утилита мониторинга процессов в реальном времени. htop – улучшенная версия top с цветным интерфейсом и удобной навигацией.

8. `find` - рекурсивный поиск файлов по имени, размеру, дате и другим критериям.

Примеры:

- `find %каталог -type d -print` - выведет все директории и под-директории в указанном каталоге.
- `find %каталог -name %имя -print` - выведет все файлы с указанным именем в текущем и под-каталогах.

9. Можно, с помощью команды `grep`. Синтаксис: `grep "текст" /путь/`.

10. С помощью команды `df`.

11. С помощью команды `du`.

12. Найти его номер PID с помощью команды `ps`, `grep` и удалить командой `kill -KILL PID`.

## Вывод

---

В результате выполнения лабораторной работы я ознакомился с инструментами поиска файлов и фильтрации текстовых данных, приобрел практические навыки по управлению процессами и по проверке использования диска.