ОТЧЕТ О ВЫПОЛНЕНИИ ЛАБОРАТОРНОЙ РАБОТЫ №6

*дисциплина: Операционные системы*

Лихтенштейн Алина Алексеевна

Содержание

# 1 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.

# 2 Задачи

1. Осуществим вход в систему, используя соответствующее имя пользователя.
2. Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишем в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
3. Выведем имена всех файлов из file.txt, имеющих расширение .conf, после чего запишем их в новый текстовой файл conf.txt.
4. Определим, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c. Предложим несколько вариантов, как это сделать.
5. Выведем на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.
6. Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.
7. Удалим файл ~/logfile.
8. Запустим из консоли в фоновом режиме редактор gedit.
9. Определим идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Узнаем, как ещё можно определить идентификатор процесса.
10. Прочтем справку (man) команды kill, после чего используем её для завершения процесса gedit.
11. Выполним команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.
12. Воспользовавшись справкой команды find, выведем имена всех директорий, имеющихся в вашем домашнем каталоге.

# 3 Теоретическое введение

## 3.1 Перенаправление ввода-вывода

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 >, >>, <, <<. Рассмотрим пример.

# Перенаправление stdout (вывода) в файл.  
# Если файл отсутствовал, то он создаётся,  
# иначе -- перезаписывается.  
   
# Создаёт файл, содержащий список дерева каталогов.  
ls -lR > dir-tree.list  
1>filename  
# Перенаправление вывода (stdout) в файл "filename".  
1>>filename  
# Перенаправление вывода (stdout) в файл "filename",  
# файл открывается в режиме добавления.  
2>filename  
# Перенаправление stderr в файл "filename".  
2>>filename  
# Перенаправление stderr в файл "filename",  
# файл открывается в режиме добавления.  
&>filename  
# Перенаправление stdout и stderr в файл "filename".

## 3.2 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 передастся на ввод команде 2

Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например:

ls -la |sort > sortilg\_list

вывод команды ls -la передаётся команде сортировки sort\verb, которая пишет результат в файл sorting\_list\verb. Чаще всего скрипты на Bash используются в качестве автоматизации каких-то рутинных операций в консоли, отсюда иногда возникает необходимость в обработке stdout одной команды и передача на stdin другой команде, при этом результат выполнения команды должен обработан.

## 3.3 Поиск файла

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

find путь [-опции]

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

**Примеры:** 1. Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на f:

find ~ -name "f\*" -print

Здесь ~ — обозначение вашего домашнего каталога, -name — после этой опции указывается имя файла, который нужно найти, “f\*” — строка символов, определяющая имя файла, -print — опция, задающая вывод результатов поиска на экран. 2. Вывести на экран имена файлов в каталоге /etc, начинающихся с символа p:

find /etc -name "p\*" -print

1. Найти в Вашем домашнем каталоге файлы, имена которых заканчиваются символом ~ и удалить их:

find ~ -name "\*~" -exec rm "{}" \;

Здесь опция -exec rm “{}” ; задаёт применение команды rm ко всем файлам, имена которых соответствуют указанной после опции -name строке символов. Для просмотра опций команды find воспользуйтесь командой man.

## 3.4 Фильтрация текста

Найти в текстовом файле указанную строку символов позволяет команда grep. Формат команды:

grep строка имя\_файла

Кроме того, команда grep способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом grep. **Примеры:** 1. Показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на f, в которых есть слово begin:

grep begin f\*

1. Найти в текущем каталоге все файлы, содержащих в имени «лаб»:

ls -l | grep лаб

## 3.5 Проверка использования диска

Команда df показывает размер каждого смонтированного раздела диска. Формат команды:

df [-опции] [файловая\_система]

**Пример:**

df -vi

Команда du показывает число килобайт, используемое каждым файлом или каталогом. Формат команды:

du [-опции] [имя\_файла...]

**Пример.**

du -a ~/

На afs можно посмотреть использованное пространство командой

fs quota

## 3.6 Управление задачами

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

gedit &

Будет запущен текстовой редактор gedit в фоновом режиме. Консоль при этом не будет заблокирована. Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач. Для завершения задачи необходимо выполнить команду

kill %номер задачи

## 3.7 Управление процессами

Любой команде, выполняемой в системе, присваивается *идентификатор процесса (process ID)*. Получить информацию о процессе и управлять им, пользуясь идентификатором процесса, можно из любого окна командного интерпретатора.

## 3.8 Получение информации о процессах

Команда ps используется для получения информации о процессах. Формат команды:

ps [-опции]

Для получения информации о процессах, управляемых вами и запущенных (работающих или остановленных) на вашем терминале, используйте опцию aux. **Пример:**

ps aux

Для запуска команды в фоновом режиме необходимо в конце командной строки указать знак & (амперсанд). Пример работы, требующей много машинного времени для выполнения, и которую целесообразно запустить в фоновом режиме:

find /var/log -name "\*.log" -print > l.log &

# 4 Выполнение лабораторной работы

* 1. Осуществим вход в систему, используя соответствующее имя пользователя (рис. [[1](#fig:001)])



Figure 1: вход в систему

1. Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишем в этот же файл названия файлов, содержащихся в вашем домашнем каталоге (рис. [[2](#fig:002)])

Figure 2: запись названий файлов в файл

Figure 2: запись названий файлов в файл

1. Выведем имена всех файлов из file.txt, имеющих расширение .conf, после чего запишем их в новый текстовой файл conf.txt (рис. [[3](#fig:003)]) (рис. [[4](#fig:004)])

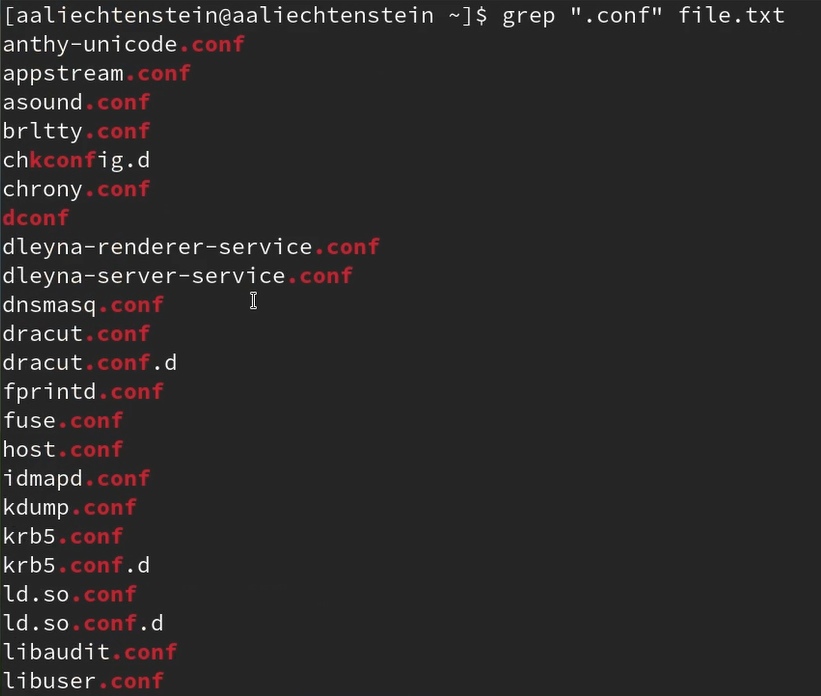


Figure 3: вывод имен файлов с расширением .conf

Figure 4: запись в новый текстовый файл

Figure 4: запись в новый текстовый файл

1. Определим, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c. Предложим несколько вариантов, как это сделать (рис. [[5](#fig:006)])

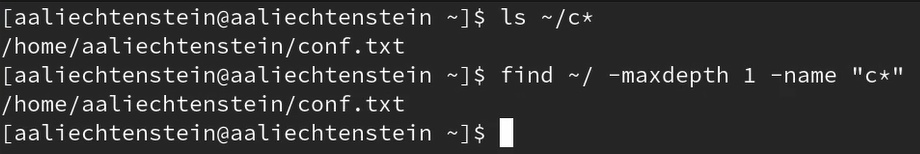


Figure 5: использование команд ls ~/C*, find ~/ -maxdepth 1 -name “c*”

1. Выведем на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h (рис. [[6](#fig:007)]) (рис. [[7](#fig:008)])

Figure 6: команда

Figure 6: команда



Figure 7: вывод команды

1. Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log (рис. [[8](#fig:009)])

Figure 8: запись файлов, начинающихся с log в файл ~/logfile

Figure 8: запись файлов, начинающихся с log в файл ~/logfile

1. Удалим файл ~/logfile (рис. [[9](#fig:010)])

Figure 9: удаление файла

Figure 9: удаление файла

1. Запустим из консоли в фоновом режиме редактор gedit (рис. [[10](#fig:011)])

Figure 10: запуск gedit

Figure 10: запуск gedit

1. Определим идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Узнаем, как ещё можно определить идентификатор процесса (рис. [[11](#fig:012)])

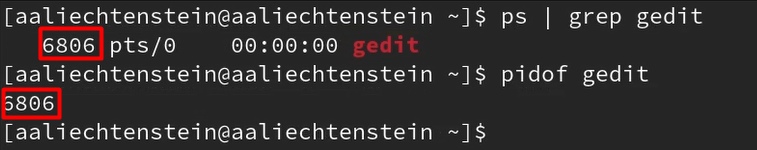


Figure 11: запуск gedit

1. Прочтем справку (man) команды kill, после чего используем её для завершения процесса gedit (рис. [[12](#fig:013)]) (рис. [[13](#fig:014)])

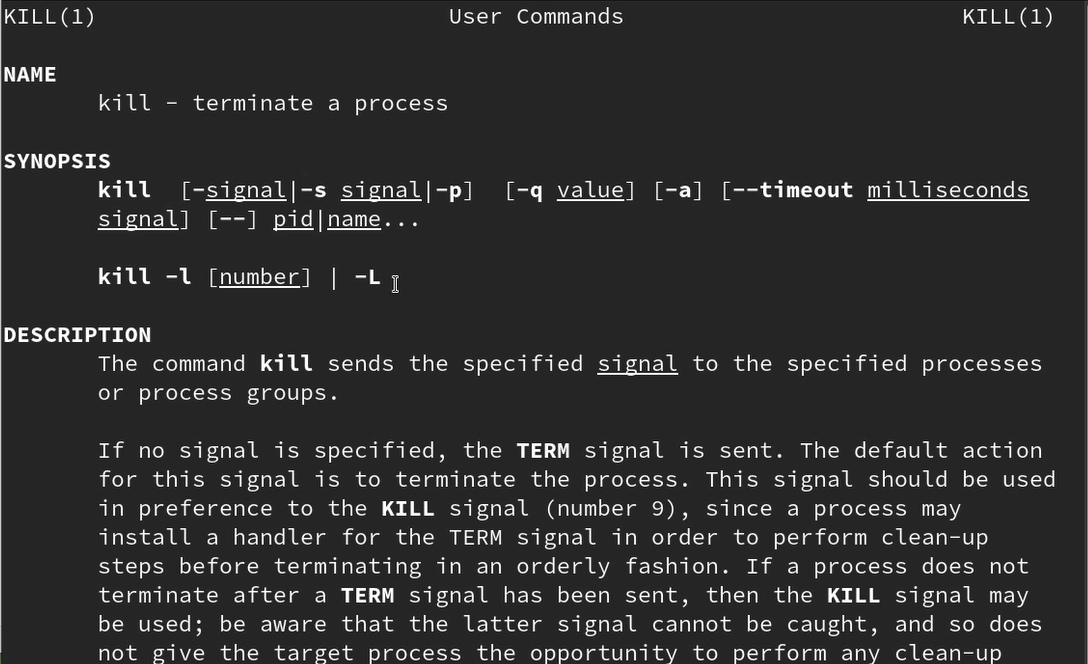


Figure 12: команда man kill

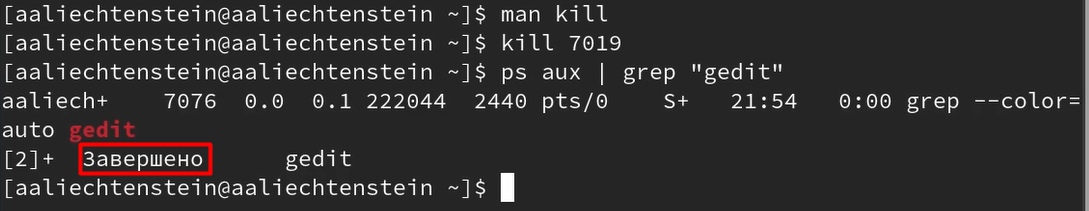


Figure 13: команда kill

1. Выполним команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man (рис. [[14](#fig:015)]) (рис. [[15](#fig:016)]) (рис. [[16](#fig:017)]) (рис. [[17](#fig:018)])

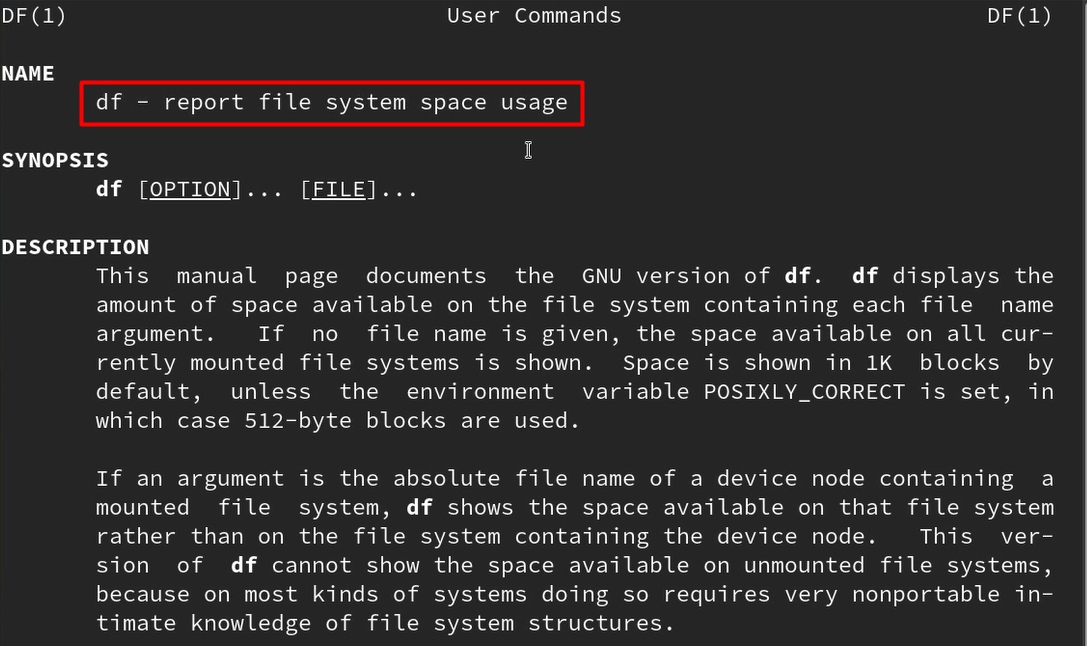


Figure 14: информация о команде df

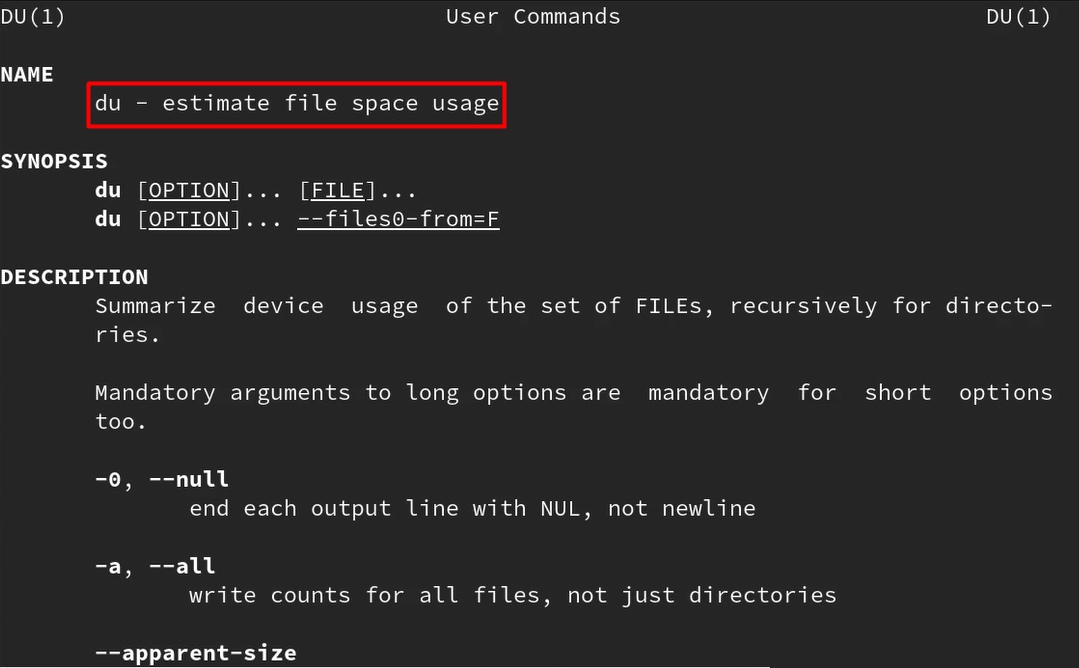


Figure 15: информация о команде du

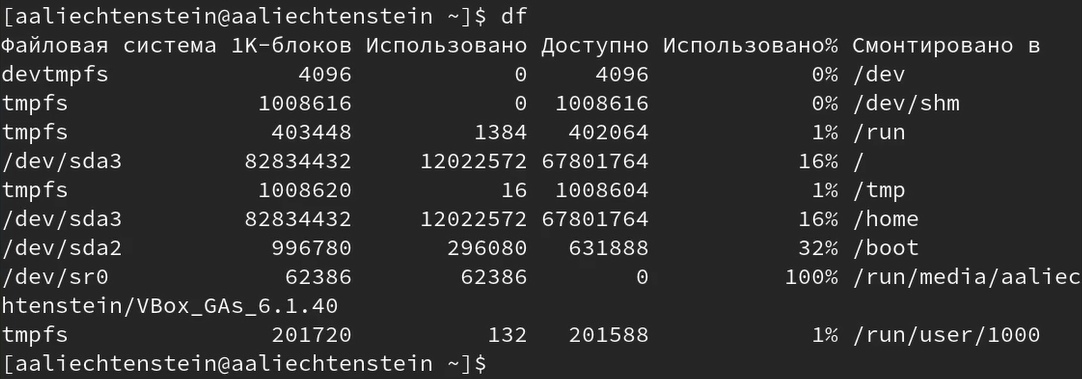


Figure 16: выполнение команды df

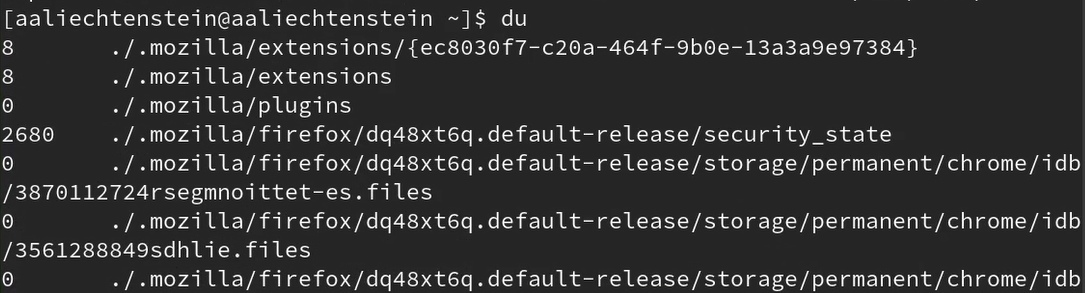


Figure 17: выполнение команды du

1. Воспользовавшись справкой команды find, выведем имена всех директорий, имеющихся в вашем домашнем каталоге (рис. [[18](#fig:019)]) (рис. [[19](#fig:020)])

Figure 18: команда

Figure 18: команда

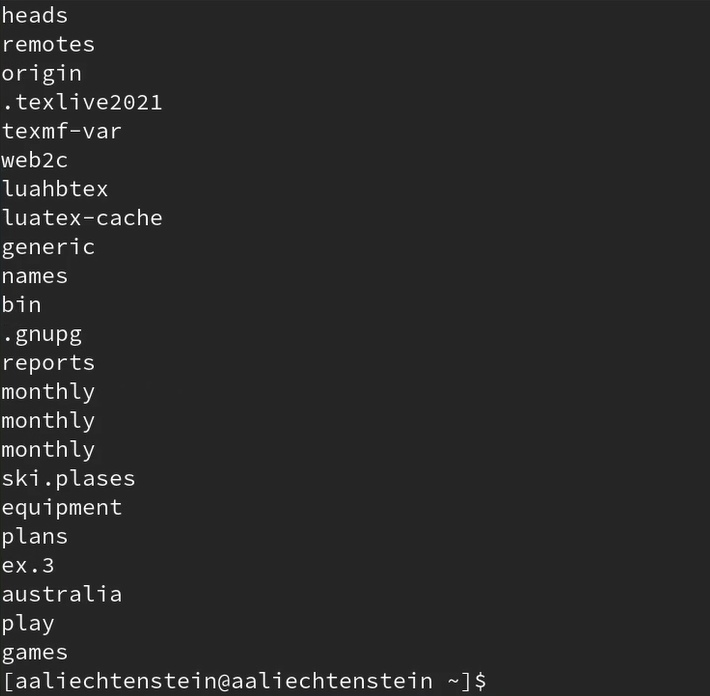


Figure 19: вывод команды

# 5 Выводы

В процессе работы мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. Были приобретены практические навыки по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.