Отчёт по лабораторной работе 8

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Чуева Злата Станиславовна

Содержание

[Цель работы 1](#_Toc1)

[Задание 1](#_Toc2)

[Теоретическое введение 1](#_Toc3)

[Выполнение лабораторной работы 2](#_Toc4)

[Выводы 2](#_Toc5)

[Контрольные вопросы 2](#_Toc6)

[Список литературы 4](#_Toc7)

# 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. Ввести команды лабораторной работы

2. Ответить на контрольные вопросы

# Теоретическое введение

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 >, >>, <, <<.

# Выполнение лабораторной работы

1. Записать в файл file.txt названия файлов, содержащихся в каталоге /etc. Дописать в этот же файл названия файлов, содержащихся в домашнем каталоге. (см Рис 1)

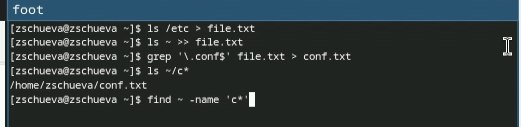


Рис 1

1. Вывести имена всех файлов из file.txt, имеющих расширение .conf, после чего записать их в новый текстовой файл conf.txt. (см Рис 1)
2. Определить, какие файлы в домашнем каталоге имеют имена, начинавшиеся с символа c. Использовать несколько вариантов, как это сделать. (см Рис 1)
3. Вывести на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h. (см Рис 2 и Рис 3)

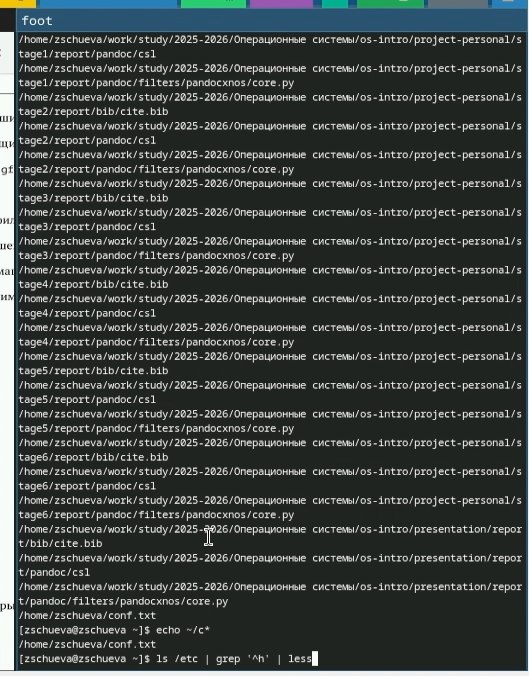


Рис 2

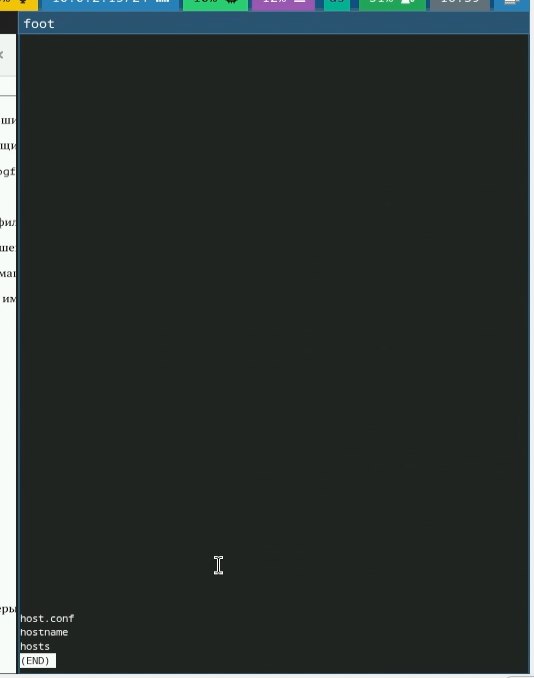


Рис 3

1. Запустить в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. (см Рис 4)

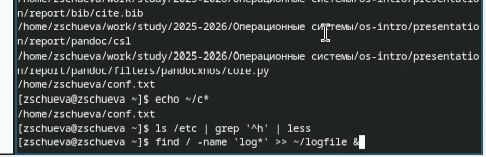


Рис 4

1. Удалить файл ~/logfile. (см Рис 5)

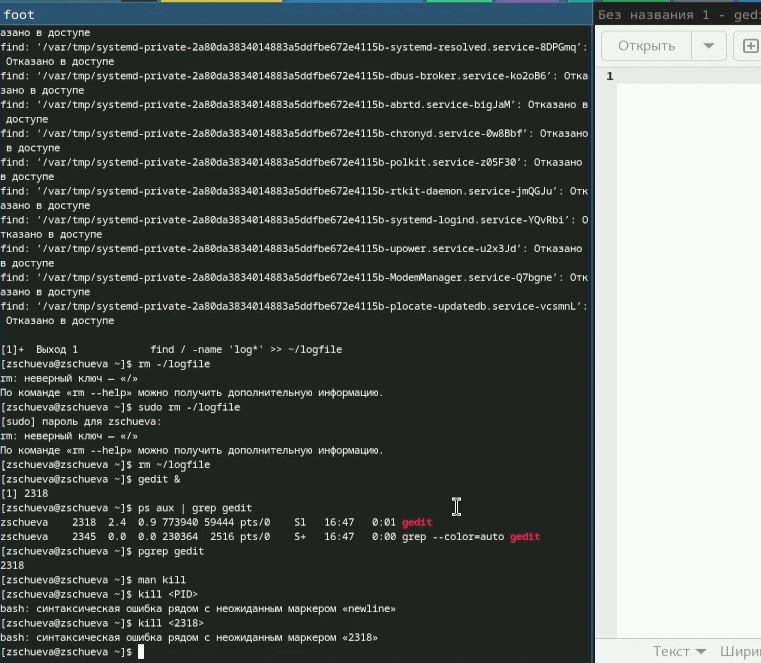


Рис 5

1. Запустить из консоли в фоновом режиме редактор gedit. (см Рис 5)
2. Определить идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. (см Рис 5)
3. Прочтать справку (man) команды kill, после чего использовать её для завершения процесса gedit. (см Рис 5 и Рис 6)

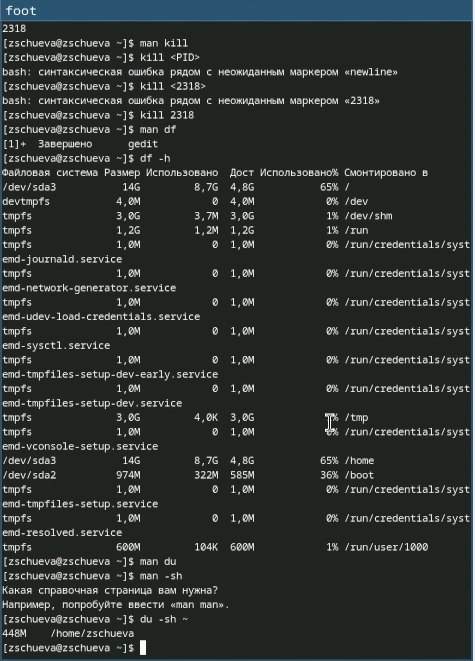
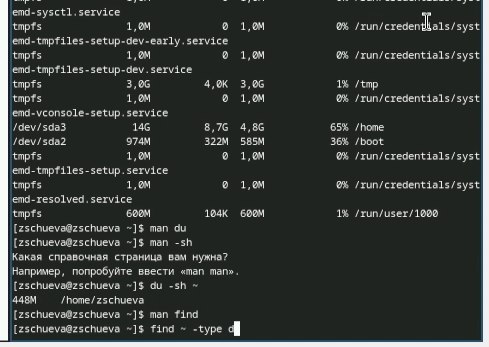


Рис 6

1. Выполнить команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man. (см Рис 6)
2. Воспользовавшись справкой команды find, вывести имена всех директорий, имеющихся в домашнем каталоге. (см Рис 7)



# Выводы

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.

# Контрольные вопросы

1. Потоки ввода-вывода: основными потоками ввода-вывода являются стандартный ввод (stdin), стандартный вывод (stdout) и стандартный вывод ошибок (stderr).

2.

> используется для перенаправления вывода команды в файл, при этом файл будет перезаписан. Если файл уже существует, его содержимое будет удалено.

>> используется для добавления вывода команды в конец файла. Если файл не существует, он будет создан. Содержимое файла при этом не будет удалено.

3. Конвейер (pipeline) — это механизм, позволяющий передавать вывод одной команды как ввод другой команды. В Unix-подобных системах это делается с помощью символа |. Например: ls -l | grep "txt" передаст вывод команды ls -l в команду grep.

4. Процесс — это экземпляр программы, выполняющийся в системе. Это активная единица работы, которая использует ресурсы системы (ЦП, память и т.д.). Программа — это статический набор инструкций, хранящихся на диске, тогда как процесс — это динамическое состояние выполнения программы.

5.

PID (Process ID) — это уникальный идентификатор процесса в системе.

GID (Group ID) — это идентификатор группы пользователей, к которой принадлежит процесс. Он используется для управления правами доступа.

6. Задачи (или процессы) — это выполняемые программы. Управлять ими можно с помощью команды kill, которая позволяет завершать процессы по их PID, а также с помощью команд ps, top, htop для просмотра текущих процессов.

7. Утилиты top и htop:

top: Это утилита для мониторинга процессов в реальном времени. Она отображает информацию о текущих процессах, их использовании ресурсов (ЦП, памяти и т.д.).

htop: Это более удобная и интерактивная версия top с цветной графикой и возможностью управления процессами с помощью клавиатуры. Htop позволяет легко сортировать процессы и управлять ими.

8. Команда поиска файлов: Основной командой для поиска файлов является find. Она позволяет искать файлы по имени, размеру, времени изменения и другим критериям.

Примеры использования:

• find /home/user -name "\*.txt" — ищет все файлы с расширением .txt в каталоге [/home](tg://bot_command?command=home)[/user](tg://bot_command?command=user).

• find / -size +100M — ищет файлы размером более 100 МБ.

9. Поиск файла по контексту: можно найти файл по содержимому с помощью команды grep. Например:

• grep -rl "искомый текст" /path/to/search — ищет все файлы, содержащие "искомый текст", начиная с указанного каталога.

10. Определение объема свободной памяти на жестком диске: Для этого используется команда df. Например:

• df -h — покажет информацию о файловых системах с размером в удобном для чтения формате.

11. Определение объема домашнего каталога: Для этого можно использовать команду du. Например:

• du -sh ~ — покажет общий объем занимаемого пространства вашим домашним каталогом.

12. Удаление зависшего процесса: Для удаления зависшего процесса можно использовать команду kill. Сначала нужно найти PID зависшего процесса с помощью ps или top, а затем выполнить:

• kill <PID> — для завершения процесса.

Если процесс не реагирует, можно использовать принудительное завершение:

• kill -9 <PID> — принудительно завершает процесс.

# Список литературы