Отчет по лабораторной работе №6:

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Федорова Наталия Артемовна

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 Задание

1. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
2. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовой файл conf.txt.
3. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать.
4. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.
5. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.
6. Удалите файл ~/logfile.
7. Запустите из консоли в фоновом режиме редактор gedit.
8. Определите идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?
9. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.
10. Выполните команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.
11. Воспользовавшись справкой команды find, выведите имена всех директорий, имеющихся в вашем домашнем каталоге.

# 3 Теоретическое введение

Приведём основные команды Unix, которые мы будем использовать в процессе выполнения лабораторной работы №6.

В табл. tbl. 1 приведен краткий справочник по командам терминала Unix-систем.

Таблица 1: Краткий справочник по командам терминала GNU Linux

| Имя команды | Описание команды |
| --- | --- |
| команда 1 | команда 2 | Конвейер |
| find путь [-опции] | Поиск файла |
| grep строка имя\_файла | Фильтрация текста |
| df [-опции] [файловая\_система] | Показывает размер каждого смонтированного раздела диска |
| du [-опции] [имя\_файла...] | Показывает число килобайт, используемое каждым файлом или каталогом |
| kill %номер задачи | Завершение задачи |
| ps [-опции] | Получение информации о процессах |

# 4 Выполнение лабораторной работы

1. Запиcала в файл file.txt названия файлов, содержащихся в каталоге /etc, используя команду ls -lR /etc > file.txt. В этот же файл добавила названия файлов, содержащихся в моем домашнем каталоге с помощью ls -lR >> file.txt. После с помощью команды cat file.txt проверяю содержимое (рис. 1).

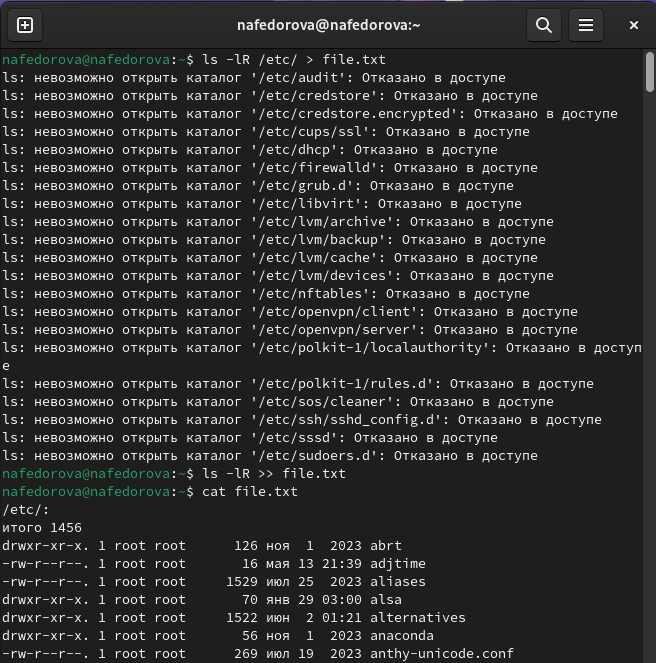


Рис. 1: Запись в файл названий других файлов

1. Вывожу имена всех файлов из file.txt, имеющих расширение .conf, после чего записала их в новый текстовой файл conf.txt. Для этого ввела команду cat file.txt | grep .conf > conf.txt(рис. 2).

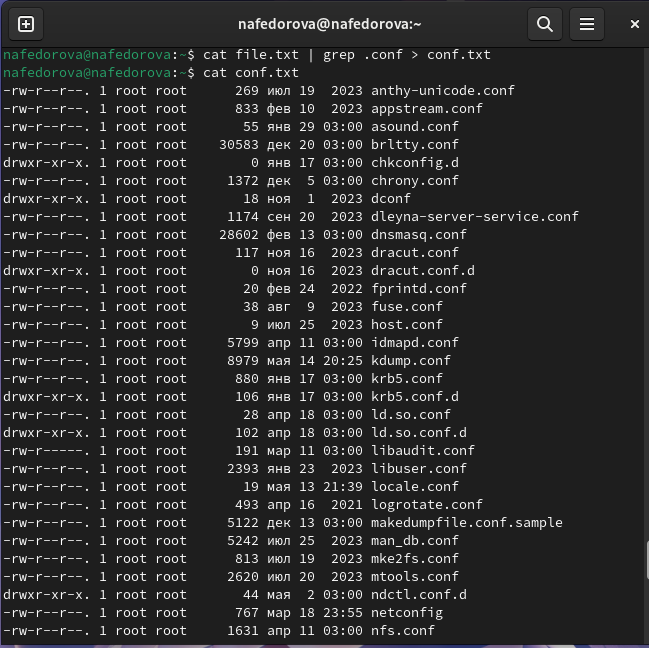


Рис. 2: Вывод имен файлов, имеющих расширение .conf

1. Определяю, какие файлы в моем домашнем каталоге имеют имена, начинавшиеся с символа ‘c’. Сначала использую вариант ls | grep c\*, а потом нахожу с помощью find /etc -name "h\*" -print (рис. 3).

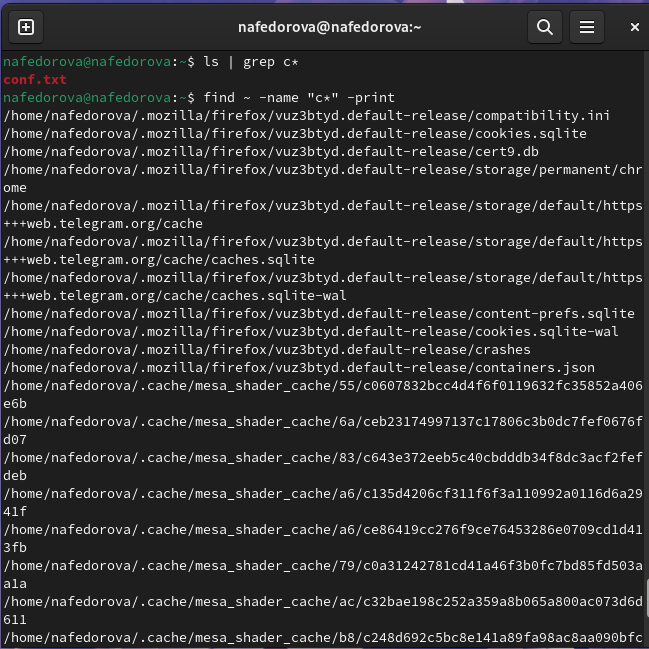


Рис. 3: Поиск файлов, начинающие с “с”, в домашнем каталоге

1. Вывожу на экран имена файлов из каталога /etc, начинающиеся с символа “h”, используя команду find ~ -name "c\*" -print (рис. 4).

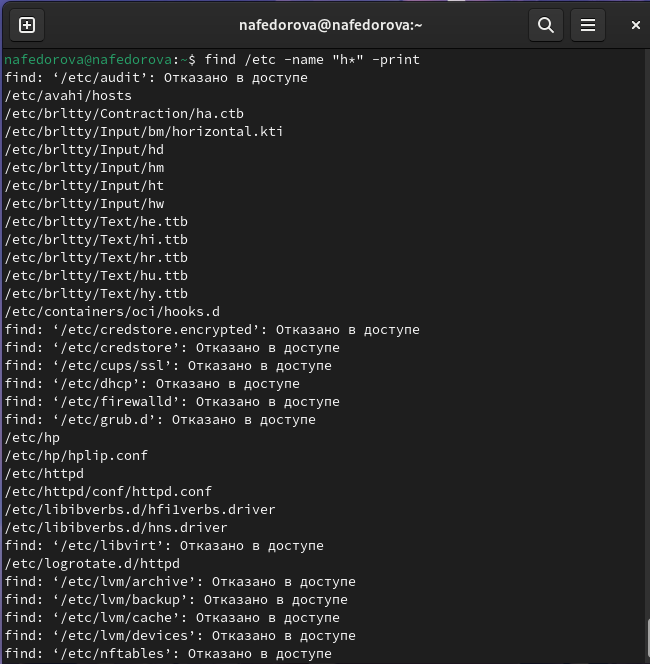


Рис. 4: Поиск файлов, начинающие с “h”, в каталоге /etc

1. Запускаю в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с “log”. Для этого использую find ~ -name "log\*" -print > logfile &. (рис. 5).

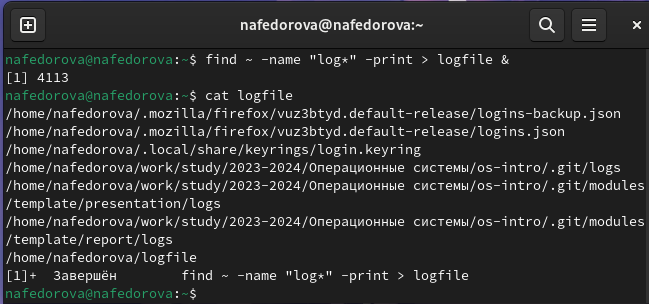


Рис. 5: Запись в файлов имен, начинающие с “log”, в фоновом режиме

1. Теперь удаляю файл ~/logfile, используя rm logfile (рис. 6).

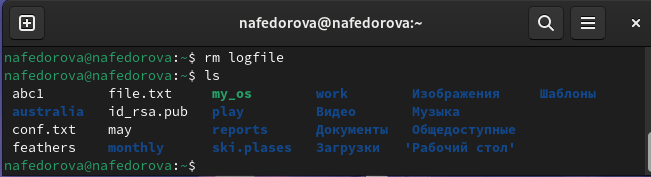


Рис. 6: Удаление файла logfile

1. Запускаю из консоли в фоновом режиме редактор gedit с помощью команды gedit & (рис. 7).

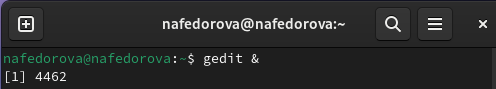


Рис. 7: Запуск в фоновом режиме редактор gedit

1. Определяю идентификатор процесса gedit, конвейер и фильтр grep. Можно использовать команду ps, чтобы вывести все процессы, а если хочу увидеть определенный процесс, то ввожу ps aux | grep gedit. В моем случае идентификатор процесса gedit равен 4462 (рис. 8).

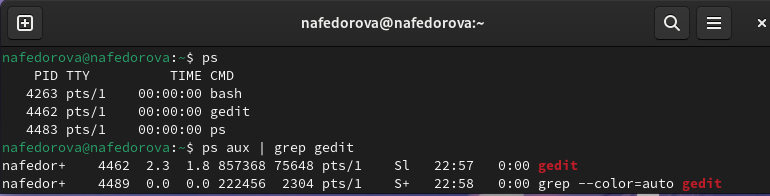


Рис. 8: Определение идентификатора процесса

1. Читаю справку (man) команды kill, после чего использую её для завершения процесса gedit (рис. 9) (рис. 10).

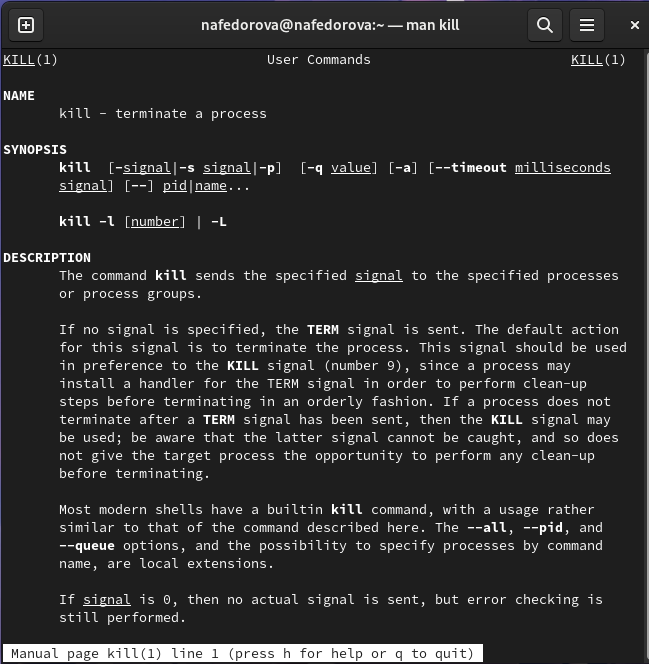


Рис. 9: Справка команды kill



Рис. 10: Завершение процесса gedit

1. Читаю более подробную информацию о командах df и du, с помощью команды man. Выполняю команды df и du.(рис. 11) (рис. 12).

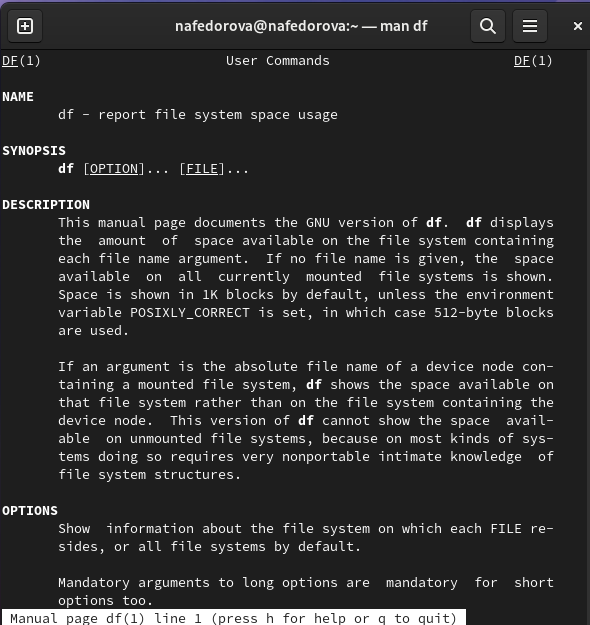


Рис. 11: Описание команды df

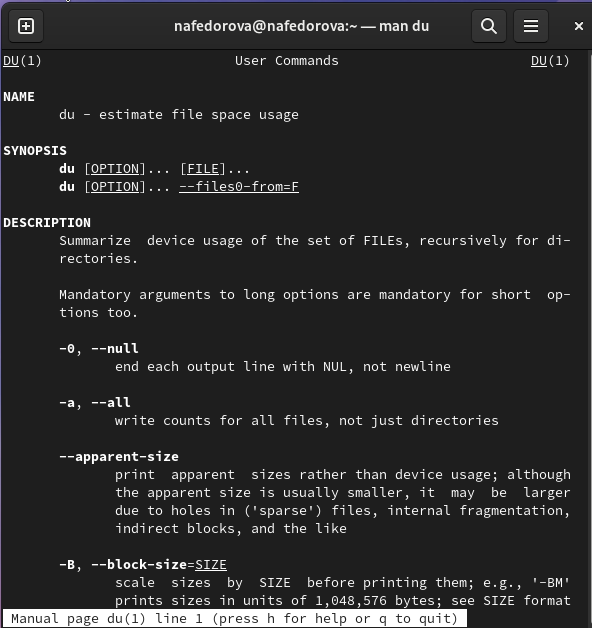


Рис. 12: Описание команды du

Команда **df** используется для отображения информации о дисковом пространстве на файловых системах, включая общий объем, использованный объем, доступное пространство и место, занятое системными файлами.

Команда **du** используется для оценки использования дискового пространства файлами и каталогами в Linux. По умолчанию du показывает использование дискового пространства для текущего каталога (рис. 13).

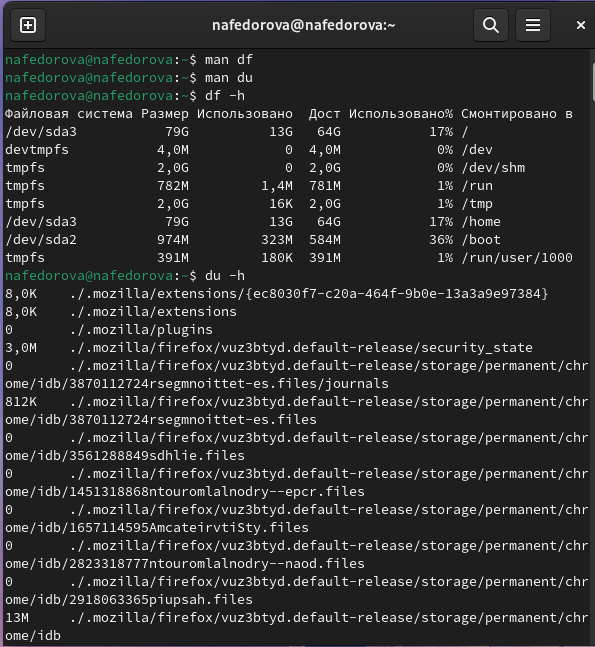


Рис. 13: Команды df и du

1. Воспользовавшись справкой команды find, вывожу имена всех директорий, имеющихся в моем домашнем каталоге, использовав команду find -type d (рис. 14) (рис. 15).

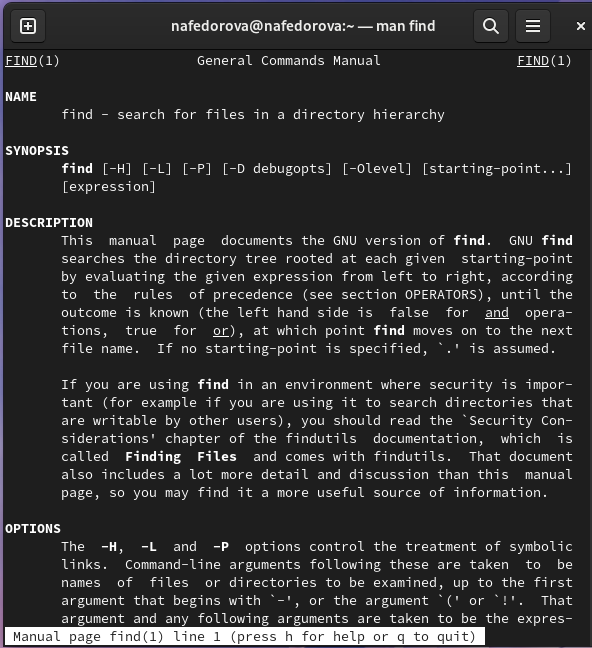


Рис. 14: Описание команды find

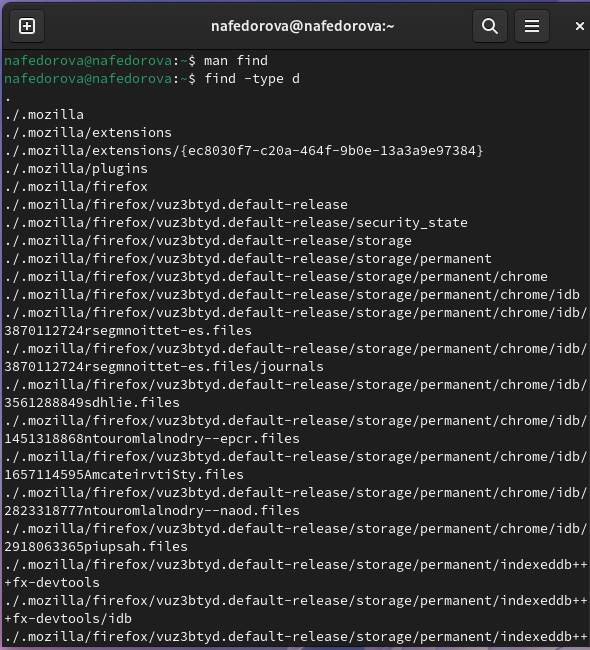


Рис. 15: Вывод всех директорий в домашнем каталоге

# 5 Контрольные вопросы

1. **Какие потоки ввода вывода вы знаете?**

Существует несколько потоков ввода-вывода в UNIX-подобных операционных системах, таких как:

* **Стандартный ввод (stdin)** - поток, через который процесс принимает ввод.
* **Стандартный вывод (stdout)** - поток, через который процесс выводит результат.
* **Стандартный поток ошибок (stderr)** - поток, через который процесс выводит сообщения об ошибках.

1. **Объясните разницу между операцией > и >>.**

Операция > используется для перенаправления вывода команды в файл, при этом файл будет перезаписан, если уже существует. Операция >> также перенаправляет вывод команды в файл, но добавляет вывод в конец файла, не перезаписывая его.

1. **Что такое конвейер?**

Конвейер (pipeline) - это механизм в UNIX, позволяющий объединить вывод одной команды с вводом другой команды без использования промежуточных файлов. Это позволяет создавать цепочки команд для выполнения сложных задач

1. **Что такое процесс? Чем это понятие отличается от программы?**

Процесс - это исполняющаяся программа во время выполнения на компьютере. Процесс включает в себя код программы, данные, которые программа использует, а также ресурсы системы, необходимые для её выполнения, такие как память, файлы и устройства ввода-вывода.

Программа, с другой стороны, это статический набор инструкций, написанный на языке программирования, который описывает алгоритм для выполнения определённой задачи. Программа становится процессом только во время выполнения, когда операционная система выделяет ей ресурсы и начинает исполнять её инструкции.

Таким образом, основное различие между процессом и программой заключается в том, что программа представляет собой статический набор инструкций, тогда как процесс - это программа во время выполнения, которая активно использует ресурсы системы для выполнения своих задач.

1. **Что такое PID и GID?**

PID (Process ID) и GID (Group ID) - это два идентификатора, используемых в операционных системах для управления процессами и пользователями.

* **PID (Process ID)** - это уникальный идентификатор, который присваивается каждому процессу в операционной системе. PID используется для отслеживания, управления и завершения процессов. Когда процесс запускается, операционная система присваивает ему уникальный PID, который остается постоянным в течение жизненного цикла процесса.
* **GID (Group ID)** - это идентификатор, который присваивается группе пользователей в операционной системе. Группы пользователей используются для управления доступом к файлам, ресурсам и другим объектам в системе. Каждый пользователь может быть членом одной или нескольких групп, и GID используется для определения, к какой группе относится пользователь.

1. **Что такое задачи и какая команда позволяет ими управлять?**

Задачи (jobs) - это процессы, запущенные в фоновом режиме в командной оболочке. Команда jobs используется для просмотра списка задач и управления ими.

1. **Найдите информацию об утилитах top и htop. Каковы их функции?**

Утилиты top и htop предоставляют информацию о процессах, выполняемых в системе, и ресурсах, которые они используют. top - это стандартная утилита, а htop представляет более удобный интерфейс для мониторинга процессов.

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

Команда поиска файлов в UNIX-подобных системах - find. Она используется для поиска файлов и каталогов в указанном месте с заданными критериями. Пример использования: find /path/to/directory -name "\*.txt".

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

Да, можно найти файл по его содержанию с помощью команды grep. Пример: grep "search\_term" file.txt.

1. **Как определить объем свободной памяти на жёстком диске?**

Чтобы определить объем свободной памяти на жестком диске, можно использовать команду df -h.

1. **Как определить объем вашего домашнего каталога?**

Чтобы определить объем вашего домашнего каталога, можно воспользоваться командой du -sh ~.

1. **Как удалить зависший процесс?**

Чтобы удалить зависший процесс, можно воспользоваться командой kill -9 PID, где PID - идентификатор процесса, который нужно завершить.

# 6 Выводы

В данной лабораторной работе я ознакомились с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практические навыки: по управлению процессами(и заданиями), по проверке использования диска и обслуживанию файловых систем.