Лабораторная работа №6

НПИбд-03-23

Дмитриев Владимир Константинович

Содержание

[1 Цель работы 1](#_Toc130074359)

[2 Задание 1](#_Toc130074360)

[3 Теоретическое введение 2](#_Toc130074361)

[4 Выполнение лабораторной работы 3](#_Toc130074362)

[5 Контрольные вопросы 12](#_Toc130074363)

[6 Выводы 14](#_Toc130074364)

[Список литературы 14](#_Toc130074365)

# 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. Осуществите вход в систему, используя соответствующее имя пользователя.
2. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовой файл conf.txt.
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать.
5. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.
6. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.
7. Удалите файл ~/logfile.
8. Запустите из консоли в фоновом режиме редактор gedit. 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, имею- щихся в вашем домашнем каталоге.

# 3 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 *>, >>, <, <<*.

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.

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

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

grep

Кроме того, команда grep способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом *grep*.

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

*df <-опции>*

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

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.

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*

# 4 Выполнение лабораторной работы

На первом этапе выполнения лабораторной работы я осуществил вход в систему, используя соответствующее имя пользователя - nmdemidovich. Далее я записал в файл file.txt названия файлов, содержащихся в каталоге /etc, после чего дописал в этот же файл названия файлов, содержащихся в своём домашнем каталоге (рис. [[1](#fig:001)]).

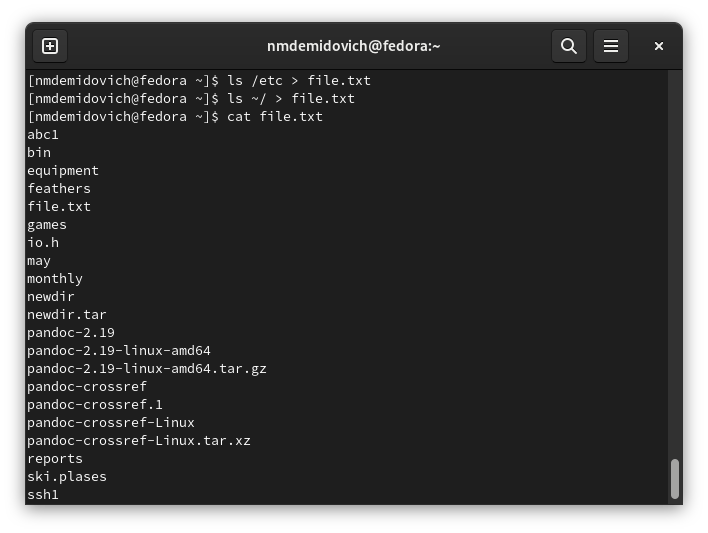


Figure 1: Добавление содержимого каталога /etc и домашнего каталога в файл file.txt

Затем я вывел имена всех файлов из file.txt, имеющих расширение .conf, после чего записал их в новый текстовой файл conf.txt (рис. [[2](#fig:002)]).

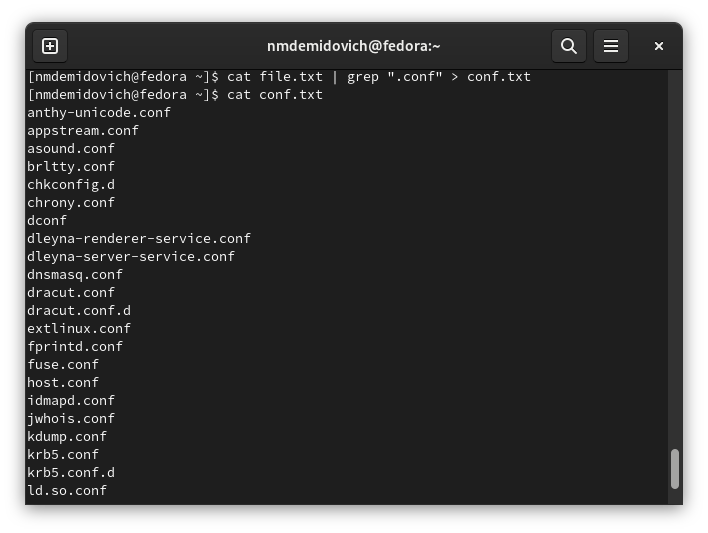


Figure 2: Новый файл conf.txt

После этого я определил, какие файлы в моём домашнем каталоге имеют имена, начинавшиеся с символа c и предложил несколько вариантов, как это сделать (рис. [[3](#fig:003)]) - (рис. [[4](#fig:004)]).



Figure 3: Первый вариант определение имён файлов, названия которых начинаются с символа “c”

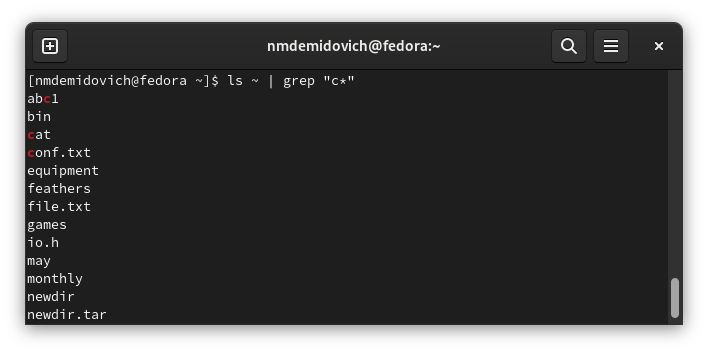


Figure 4: Второй вариант определение имён файлов, названия которых начинаются с символа “c”

Далее я вывел на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h (рис. [[5](#fig:005)])

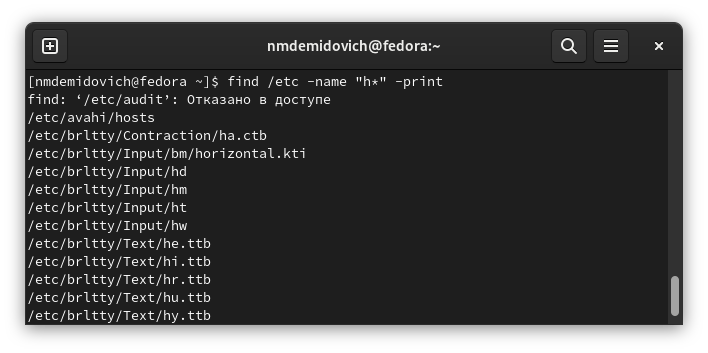


Figure 5: Вывод имён файлов, начинающихся на h

Затем я запустил в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log, после чего удалил его (рис. [[6](#fig:006)]) - (рис. [[8](#fig:008)]).

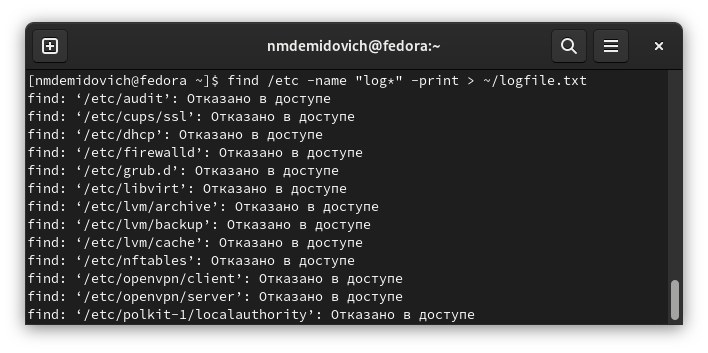


Figure 6: Запуск фонового процесса

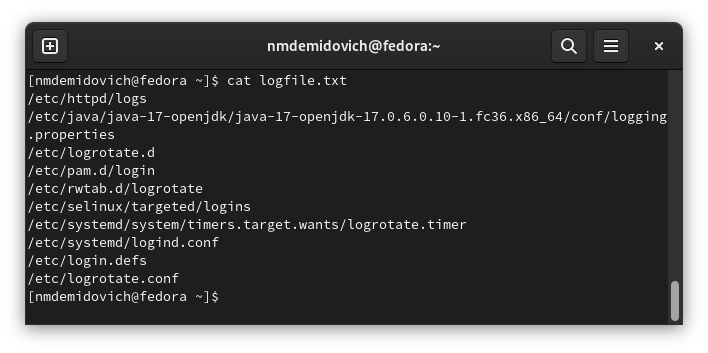


Figure 7: Новый файл logfile.txt



Figure 8: Удаление файла logfile.txt

После этого я запустил из консоли в фоновом режиме редактор gedit. И определил идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep (рис. [[9](#fig:009)]). Идентификатор процесса можно определить с помощью команд ps aux | grep “gedit”, ps -fC gedit и pidof gedit.

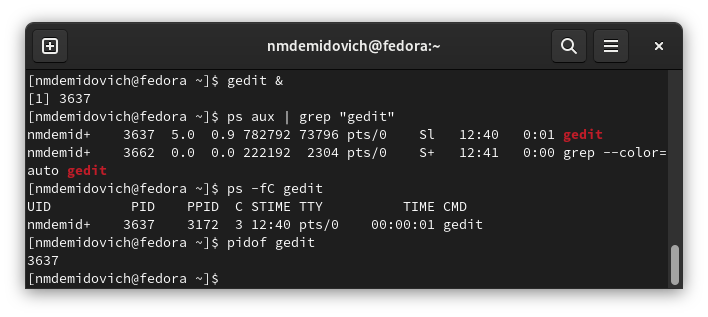


Figure 9: Команды ps aux | grep “gedit”, ps -fC gedit и pidof gedit для определения идентификатора процесса

Далее я прочёл справку (man) команды kill (рис. [[10](#fig:010)]).

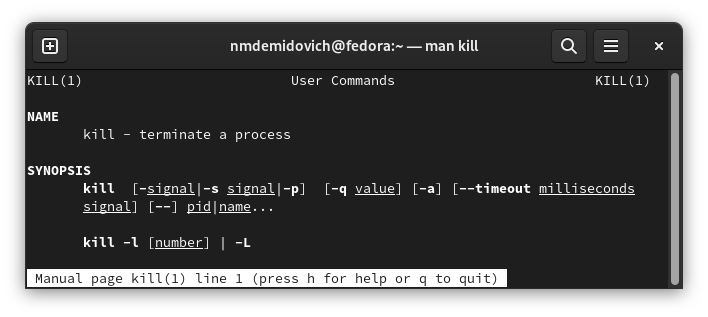


Figure 10: Справка (man) команды kill

Я не использовал её для завершение процесса, т.к он был завершен мною ранее.

Затем я выполнил команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man (рис. [[11](#fig:011)]) - (рис. [[15](#fig:015)])

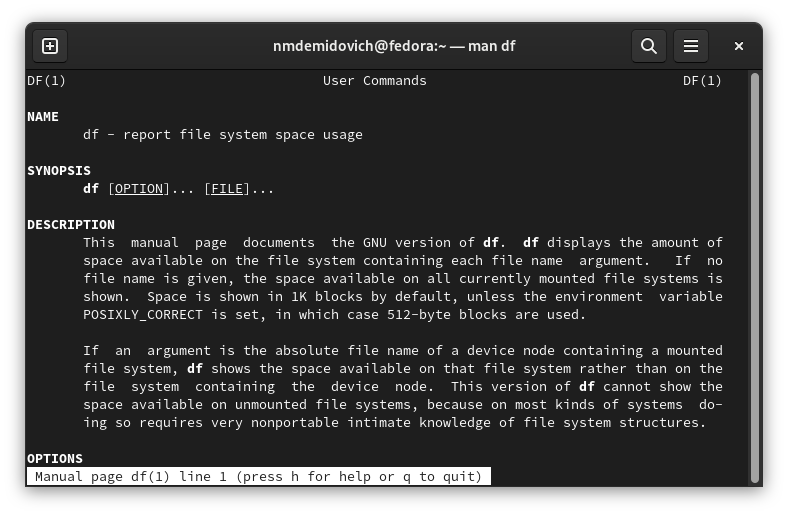


Figure 11: Справка (man) команды df

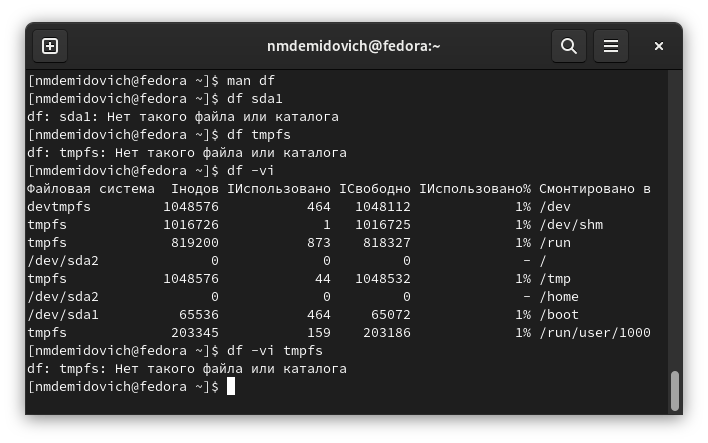


Figure 12: Выполнение команды df

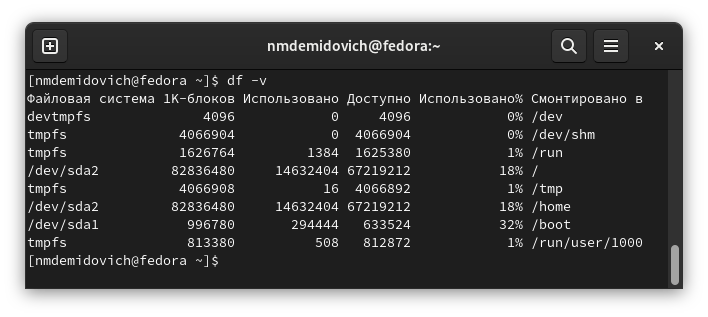


Figure 13: Выполнение команды df

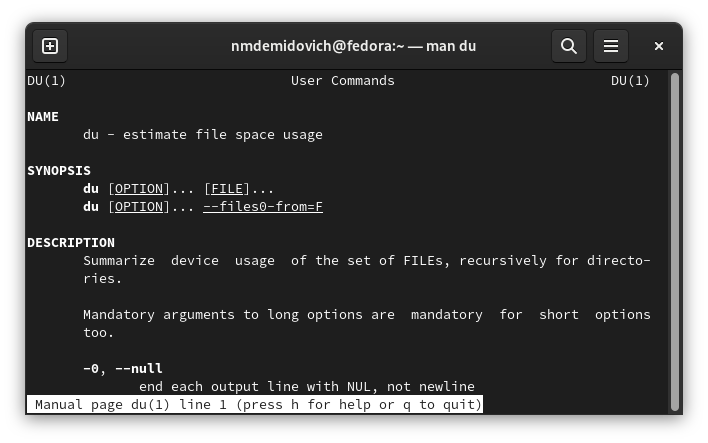


Figure 14: Справка (man) команды du

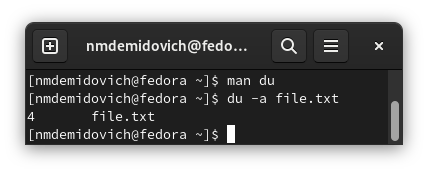


Figure 15: Выполнение команды du

И на заключительном этапе выполнения работы я, воспользовавшись справкой команды find, вывел имена всех директорий, имеющихся в моём домашнем каталоге (рис. [[16](#fig:016)]) - (рис. [[17](#fig:017)])

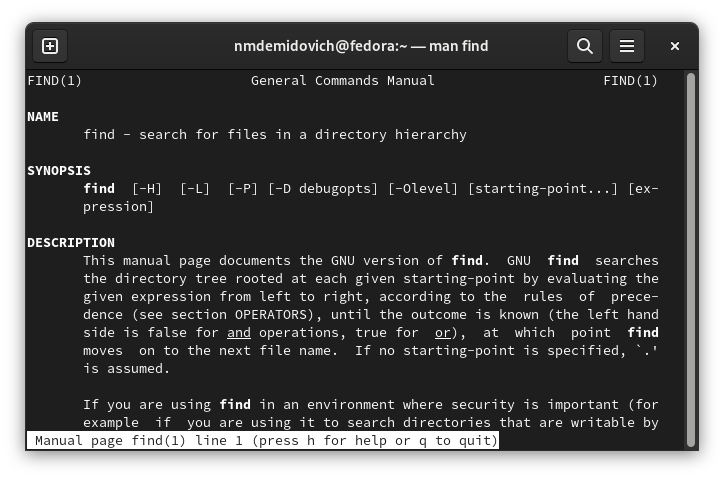


Figure 16: Справка (man) команды find

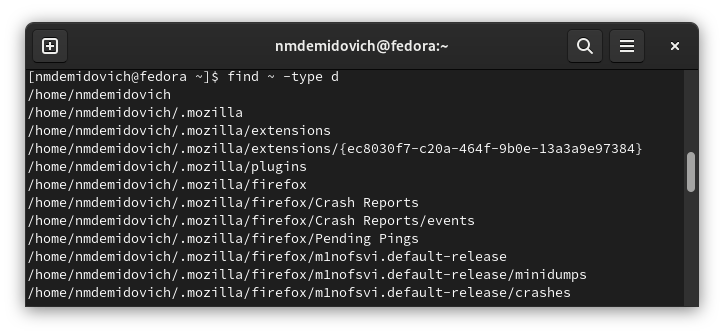


Figure 17: Выполнение команды find

# 5 Контрольные вопросы

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.

1. Объясните разницу между операцией > и >>.

>filename - Перенаправление вывода (stdout) в файл “filename”.

>>filename - Перенаправление вывода (stdout) в файл “filename”, файл открывается в режиме добавления.

1. Что такое 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

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

Компьютерная программа сама по себе — лишь пассивная последовательность инструкций. В то время как процесс — непосредственное выполнение этих инструкций.

1. Что такое PID и GID?

Идентификатор процесса (PID). Каждому новому процессу ядро присваивает уникальный идентификационный номер. В любой момент времени идентификатор процесса является уникальным, хотя после завершения процесса он может использоваться снова для другого процесса. Некоторые идентификаторы зарезервированы системой для особых процессов. Так, процесс с идентификатором 1 - это процесс инициализации init, являющийся предком всех других процессов в системе.

Идентификатор группы GID и эффективный идентификатор группы (EGID) GID - это идентификационный номер группы данного процесса. EGID связан с GID также, как EUID с UID.

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

Задачи - это то, что мы подаем на выполнение системе, какой-то процесс, который она начинает выполнять.

Команда - jobs.

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

top (table of processes) — консольная команда, которая выводит список работающих в системе процессов и информацию о них. По умолчанию она в реальном времени сортирует их по нагрузке на процессор. Программа написана для UNIX-совместимых операционных систем и опубликована под свободной лицензией GNU FDL.

htop — продвинутый монитор процессов, написанный для Linux. Он был задуман заменить стандартную программу top. Htop показывает динамический список системных процессов, список обычно выравнивается по использованию ЦПУ. В отличие от top, htop показывает все процессы в системе. Также показывает время непрерывной работы, использование процессоров и памяти. Htop часто применяется в тех случаях, когда информации даваемой утилитой top недостаточно, например при поиске утечек памяти в процессах. Htop написан на языке Си и использует для отображения библиотеку Ncurses.

1. Назовите и дайте характеристику команде поиска файлов. Приведите примеры ис- пользования этой команды.

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

find <путь> <-опции>

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

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

Для поиска файла по содержимому проще всего воспользоваться командой grep (вместо find). Пример:

grep -r строка\_поиска каталог

1. Как определить объем свободной памяти на жёстком диске?

При помощи команды df (аббревиатура от disk free) — утилита в UNIX и UNIX-подобных системах, показывает список всех файловых систем по именам устройств, сообщает их размер, занятое и свободное пространство и точки монтирования.

1. Как определить объем вашего домашнего каталога?

du -a ~

1. Как удалить зависший процесс?

Для завершения процесса нужно вызвать утилиту kill с параметром "-9".

# 6 Выводы

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

# Список литературы

Лабораторная работа №6 (Архитектура ОС)