Отчёт по лабораторной работе №6. Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Дисциплина: Операционные системы

Ганина Таисия Сергеевна, НКАбд-01-22

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. 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

1. Осуществите вход в систему, используя соответствующее имя пользователя:

Вход осуществлялся при помощи пароля от учётной записи tsganina.

1. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге (рис. [1](#fig:001), [2](#fig:002)).

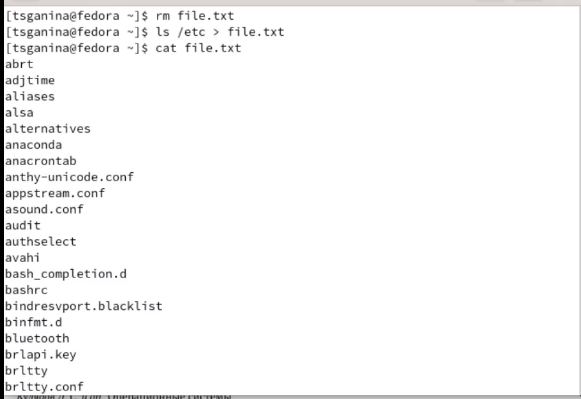


Figure 1: Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc

Figure 2: Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге

Figure 2: Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге

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

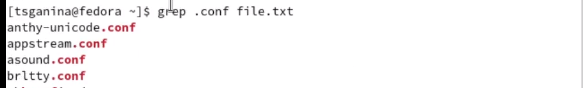


Figure 3: Файлы из file.txt, имеющие расширение .conf

Figure 4: Их запись в новый текстовый файл

Figure 4: Их запись в новый текстовый файл

1. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать (рис. [5](#fig:005), [6](#fig:006), [7](#fig:007)).

Figure 5: Первый вариант при помощи find ~ -name “c*” -print

Figure 5: Первый вариант при помощи find ~ -name “c\*” -print

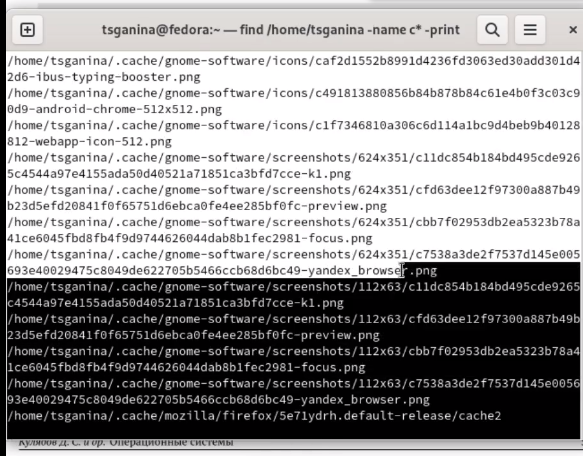


Figure 6: Результат



Figure 7: Второй вариант при помощи ls ~ | grep c\*

1. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h (рис. [8](#fig:008)).

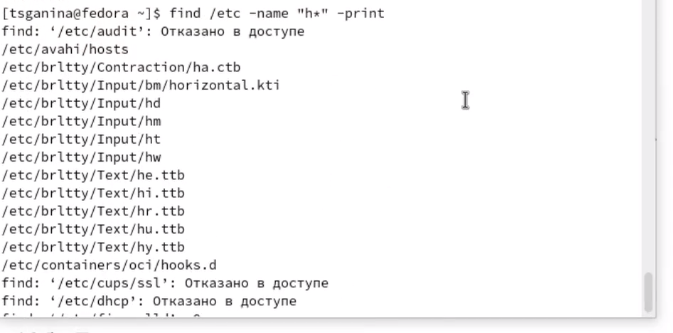


Figure 8: find /etc -name “h\*” -print

1. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log (рис. [9](#fig:009), [10](#fig:010)).

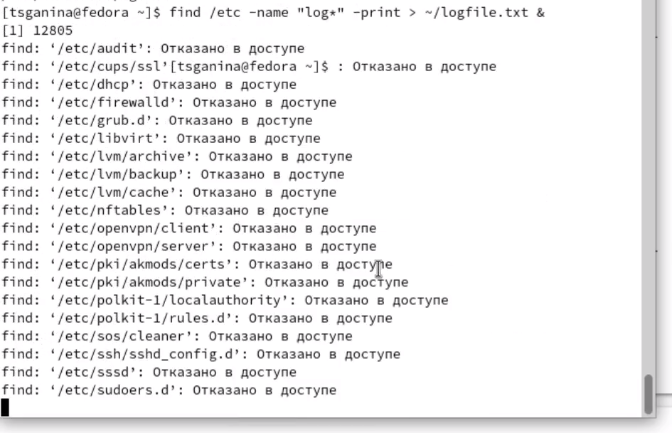


Figure 9: Процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log



Figure 10: Результат

1. Удалите файл ~/logfile (рис. [11](#fig:011)).

Figure 11: Удаление файла

Figure 11: Удаление файла

1. 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? (рис. [12](#fig:012)).



Figure 12: 1) ps aux | grep “gedit”; 2) ps -fC gedit; 3) pidof gedit

1. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit (рис. [13](#fig:013), [14](#fig:014)).

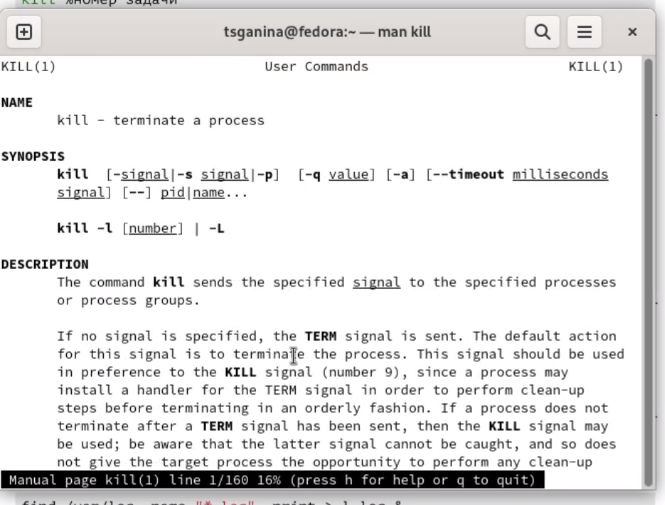


Figure 13: Справка (man) команды kill



Figure 14: Завершение процесса gedit

1. Выполните команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man (рис. [15](#fig:015), [16](#fig:016), [17](#fig:017), [18](#fig:018), [19](#fig:019)).

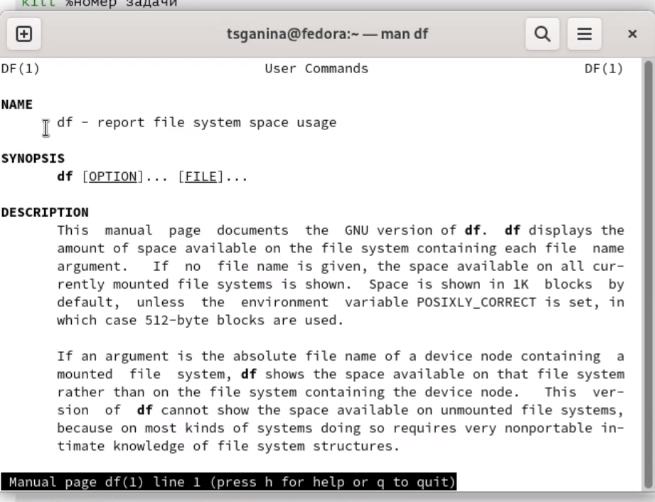


Figure 15: Справка df

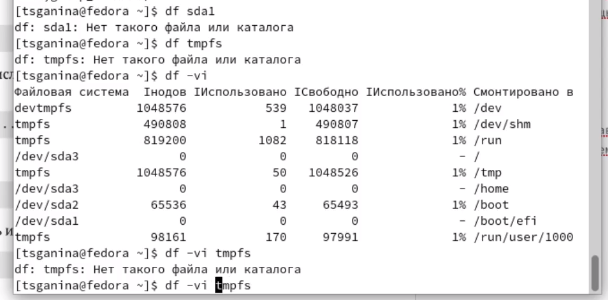


Figure 16: Результат

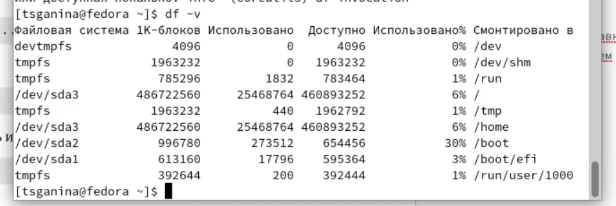


Figure 17: Результат

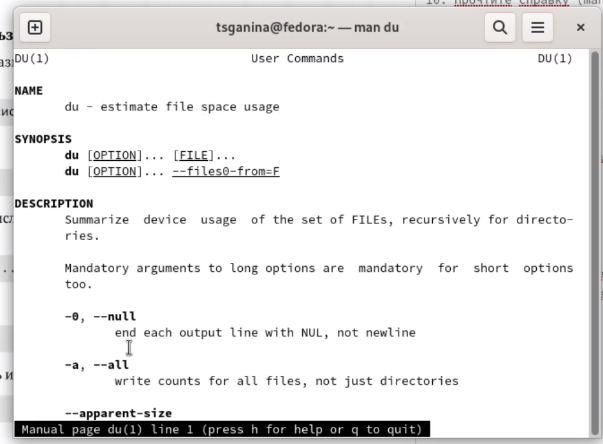


Figure 18: Справка du



Figure 19: Результат

1. Воспользовавшись справкой команды find, выведите имена всех директорий, имею- щихся в вашем домашнем каталоге (рис. [20](#fig:020), [21](#fig:021), [22](#fig:022)).

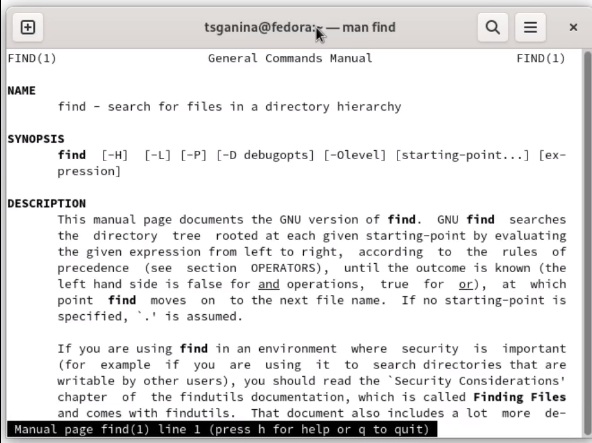


Figure 20: Справка find

Figure 21: Выполнение

Figure 21: Выполнение

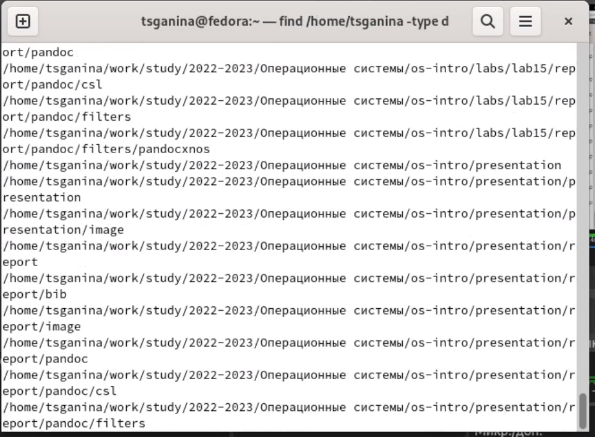


Figure 22: Результат

# 5 Ответ на 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 ознакомилась с инструментами поиска файлов и фильтрации текстовых данных и приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

# Список литературы

1. [Руководство к выполнению лабораторной работы](https://esystem.rudn.ru/pluginfile.php/1975771/mod_resource/content/4/006-lab_proc.pdf)
2. [Поиск файлов и данных](https://1cloud.ru/blog/find_data_in_linux)