Отчет по выполнению лабораторной работы №8

Операционные системы

Шихалиева Зурият Арсеновна

Содержание

# 1 Цель работы

Цель данной лабораторной работы - ознакомление с инструментами поиска файлов и фильтрации текстовых данных, а также приобретение практических навыков по управлению процессами (и заданиями), по проверке использования диска и по обслуживанию файловых систем.

# 2 Задание

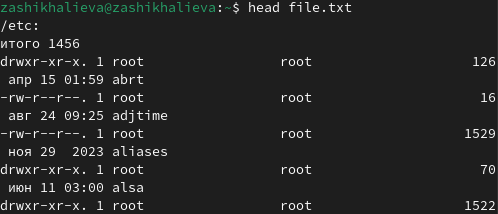
1. Осуществите вход в систему, используя соответствующее имя пользователя.
2. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.
3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после 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.
9. Определите идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?
10. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.
11. Выполните команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man.
12. Воспользовавшись справкой команды find, выведите имена всех директорий, имеющихся в вашем домашнем каталоге.

# 3 Выполнение лабораторной работы

Я вошла в систему под соотвествующим именем пользователя, открыла терминал, записала в файл file.txt названия файлов из каталога /etc с помощью перенаправления “>” (и файл создала, и записала в него то, что могло быть выведено ls -lR /etc). В файл я добавила также все файлы из подкаталогов (рис. 1).

Запись в файл{#fig:001 width=70% рис. 1}

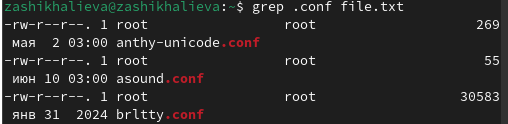
Проверила, что в файл записались нужные значения с помощью утилиты head, она выводит первые 10 строк файла на экран (рис. 2).

{#fig:002 width=70% рис. 2}

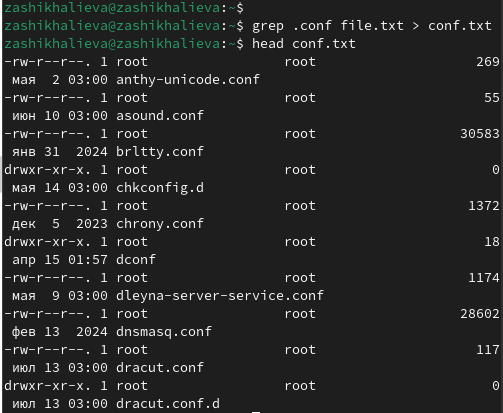
Добавила в созданный файл имена файлов из домашнего каталога, используя перенаправление “>>” в режиме добавления (рис. 3).

Добавление данных в файл{#fig:003 width=70% рис. 3}

Вывела на экран имена всех файлов, имеющих расширение “.conf” с помощью утилиты grep (рис. 4).

{#fig:004 width=70% рис. 4}

Добавила вывод прошлой команды в новый файл conf.txt с помощью перенаправления “>” (файл создается при выполнении этой команды) (рис. 5).

{#fig:005 width=70% рис. 5}

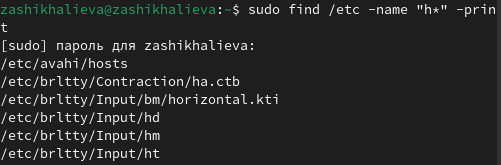
Определяю, какие файлы в домашнем каталоге начинаютcя с символа “c” с помощью утилиты find, прописываю ей в аргументах домашнюю директорию (тогда вывод относительно корневого каталога, а не домашнего будет), выбираю опцию -name (ищем по имени), и пишу маску, по которой будем искать имя, где \* - любое количество любых символов, добавляю опцию -print, чтобы мне вывелся результат (рис. 6). Но таким образом я получаю информацию даже о файлах из подкаталогов домашнего каталога.

{#fig:006 width=70% рис. 6}

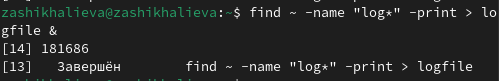
Второй способ использовать утилиту ls -lR и использовать grep, чтобы найти элементы с первым символом c. Однако этот способ не работает для поиска файлов из подкаталогов каталога (рис. 7).

Поиск файлов, начинающихся с определенного элемента{#fig:007 width=70% рис. 7}

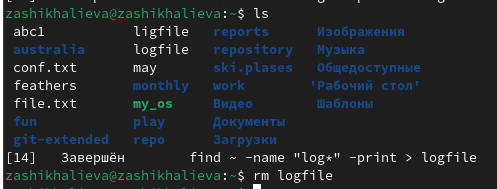
С помощью метода find, чьи опции я расписала ранее, ищу все файлы, начинающиеся с буквы “h” (рис. 8).

{#fig:008 width=70% рис. 8}

Запускаю в фоновом режиме (на это указывает символ &) процесс, который будет записывать в файл logfile (с помощью перенаправления >) файлы, имена которых начинаются с log (рис. 9).

{#fig:009 width=70% рис. 9}

Проверяю, что файл создан, удаляю его, проверяю, что файл удален (рис. 10).

{#fig:010 width=70% рис. 11}

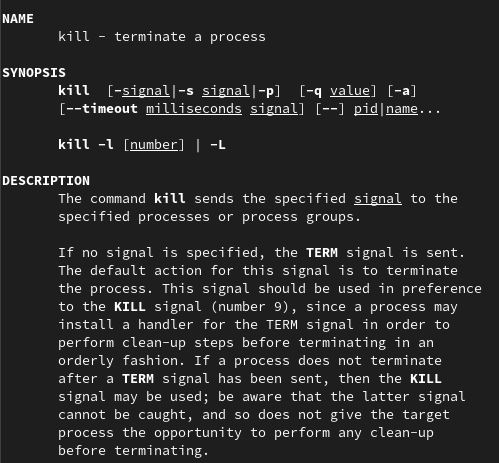
Запускаю в консоли в фоновом режиме (с помощью символа &) редактор gedit(рис. 11).

Создание фонового процесса{#fig:011 width=70% рис. 11}

С помощью утилиты ps определяю идентификатор процесса mousepad, его значение 3913 (рис. 12). Также мы можем определить идентификатор с помощью pgrep.

{#fig:012 width=70% рис. 12}

Прочитала справку команды kill (рис. 13).

{#fig:013 width=70% рис. 13}

Использую команду kill и идентификатор процесса, чтобы его удалить (рис. 14). Заметила, как у меня закрылась программа gedit.

Удаление процесса{#fig:014 width=70% рис. 14}

Прочитала документацию про функции df и du (рис. 15).

Чтение документации{#fig:015 width=70% рис. 15}

Использую утилиту df опции -iv позволяют увидеть информацию об инодах и сделать вывод читаемым, игнорируя сообщения системы о нем (рис. fig. 1). Эта утилита нам нужна, чтобы выяснить, сколько свободного места есть у нашей системы.

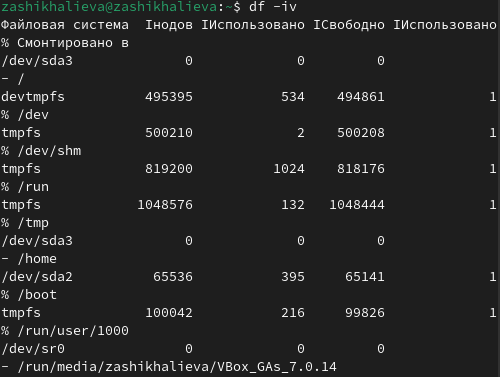
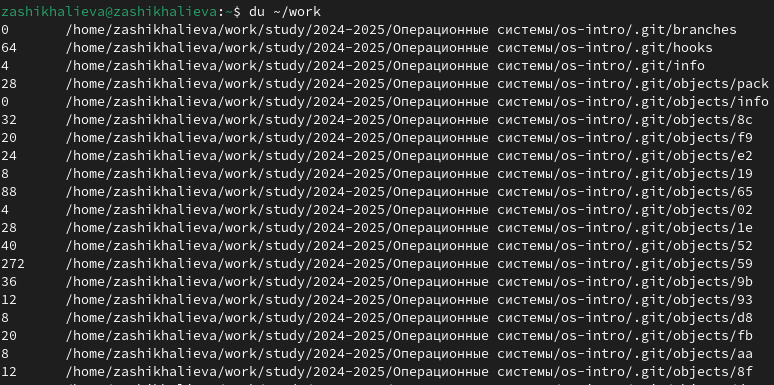
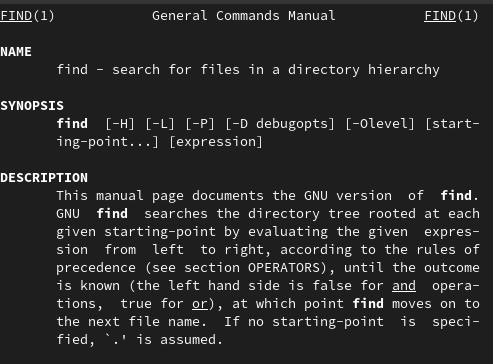


Рис. 1: Утилита df

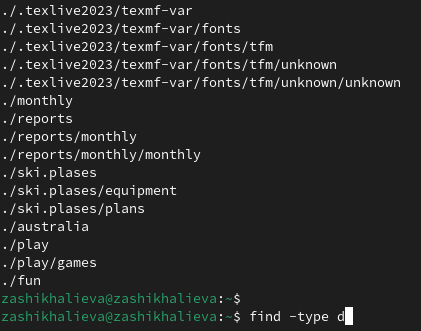
Использую утилиту du. Она нужна чтобы просмотреть, сколько места занимают файлы в определенной директории и найти самые большие из них (рис. 17).

{#fig:017 width=70% рис. 17}

Прочитала документацию о команде find (рис. 18).

{#fig:018 width=70% рис. 18}

Вывела имена всех директорий, имеющихся в моем домашнем каталоге, используя аргумент d у утилиты find опции -type, то есть указываю тип файлов, который мне нужен и этот тип Директория (рис. 19). Утилита -а позволит увидеть размер всех файлов, а не только диреткорий.

{#fig:019 width=70% рис. 19}

# 4 Выводы

В результате данной лабораторной работы я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных, а также приобрела практические навыки по управлению процессами (и заданиями), по проверке использования диска и по обслуживанию файловых систем.

# 5 Ответы на контрольные вопросы

1. 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.
2. Этот знак > - перенаправление ввода/вывода, а » - перенаправление в режиме добавления.
3. Конвейер (pipe) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.
4. Главное отличие между программой и процессом заключается в том, что программа - это набор инструкций, который позволяет ЦПУ выполнять определенную задачу, в то время как процесс - это исполняемая программа.
5. PPID - (parent process ID) идентификатор родительского процесса. Процесс может порождать и другие процессы. UID, GID - реальные идентификаторы пользователя и его группы, запустившего данный процесс.
6. Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач.
7. Команда htop похожа на команду top по выполняемой функции: они обе показывают информацию о процессах в реальном времени, выводят данные о потреблении системных ресурсов и позволяют искать, останавливать и управлять процессами. У обеих команд есть свои преимущества. Например, в программе htop реализован очень удобный поиск по процессам, а также их фильтрация. В команде top это не так удобно — нужно знать кнопку для вывода функции поиска. Зато в top можно разделять область окна и выводить информацию о процессах в соответствии с разными настройками. В целом top намного более гибкая в настройке отображения процессов.
8. Команда find - это одна из наиболее важных и часто используемых утилит системы Linux. Это команда для поиска файлов и каталогов на основе специальных условий. Ее можно использовать в различных обстоятельствах, например, для поиска файлов по разрешениям, владельцам, группам, типу, размеру и другим подобным критериям. Утилита find предустановлена по умолчанию во всех Linux дистрибутивах, поэтому вам не нужно будет устанавливать никаких дополнительных пакетов. Это очень важная находка для тех, кто хочет использовать командную строку наиболее эффективно.

Команда find имеет такой синтаксис: find [папка] [параметры] критерий шаблон [действие] Пример: find /etc -name “p\*” -print

1. Можно. Через следующую команду: find / -type f -exec grep -H ‘текстДляПоиска’ {} ;
2. Это можно определить с помощью команды df -h.
3. Это можно определить с помощью команды du -s.
4. Это можно сделать с помощью команды kill% номер задачи.