

Отчёт по лабораторной работе №6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Аскеров Александр Эдуардович

Содержание

1	Цель работы	4
2	Задание	5
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Контрольные вопросы	12
6	Выводы	17

Список иллюстраций

4.1	Запись в файл file.txt	8
4.2	Дописывание в файл file.txt	8
4.3	Имена файлов из file.txt	8
4.4	Запись из file.txt в conf.txt	9
4.5	Файлы в домашнем каталоге, начинающиеся с “с” (двумя способами)	9
4.6	Файлы в каталоге /etc, начинающиеся с “с”	9
4.7	Запуск фонового процесса	9
4.8	Удаление файла logfile	10
4.9	Фоновый запуск gedit	10
4.10	Определение идентификатора процесса gedit (двумя способами) .	10
4.11	Использование команды kill	10
4.12	Команда df	11
4.13	Команда du	11
4.14	Имена директорий из домашнего каталога	11

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 текстовый файл `conf.txt`.
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинающиеся с символа `s`? 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 директо-
рий, имеющихся в вашем домашнем каталоге.

3 Теоретическое введение

Перенаправление ввода-вывода – возможность командной оболочки ряда операционных систем перенаправлять стандартные потоки в определённое пользователем место.

Команда `find` используется для поиска и отображения на экране имён файлов, соответствующих заданной строке символов. Формат команды: `find путь [-опции]`

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.

4 Выполнение лабораторной работы

1. Осуществляем вход в систему, используя соответствующее имя пользователя.
2. Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc.

```
[aeaskerov@fedora ~]$ ls -a /etc > file.txt  
[aeaskerov@fedora ~]$
```

Рис. 4.1: Запись в файл file.txt

Допишем в этот же файл названия файлов, содержащихся в домашнем каталоге.

```
[aeaskerov@fedora ~]$ ls -a ~ >> file.txt  
[aeaskerov@fedora ~]$
```

Рис. 4.2: Дописывание в файл file.txt

3. Выведем имена всех файлов из file.txt, имеющих расширение .conf, после чего запишем их в новый текстовый файл conf.txt.

```
[aeaskerov@fedora ~]$ cat file.txt | grep .conf$  
anthy-unicode.conf  
appstream.conf  
asound.conf  
brltty.conf
```

Рис. 4.3: Имена файлов из file.txt


```
[aeaskerov@fedora ~]$ cat file.txt | grep .conf$ >> conf.txt
[aeaskerov@fedora ~]$
```

Рис. 4.4: Запись из file.txt в conf.txt

4. Определим, какие файлы в домашнем каталоге имеют имена, начинавшиеся с символа с?

```
[aeaskerov@fedora ~]$ find ~ -maxdepth 1 -name "c*"
/home/aeaskerov/conf.txt
[aeaskerov@fedora ~]$ ls | grep -w c*
conf.txt
[aeaskerov@fedora ~]$
```

Рис. 4.5: Файлы в домашнем каталоге, начинающиеся с “с” (двумя способами)

5. Выведем на экран (постранично) имена файлов из каталога /etc, начинающиеся с символа h.

```
[aeaskerov@fedora ~]$ find /etc -maxdepth 1 -name h*
/etc/hp
/etc/httpd
```

Рис. 4.6: Файлы в каталоге /etc, начинающиеся с “с”

6. Запустим в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.

```
[aeaskerov@fedora ~]$ find /etc -maxdepth 1 -name "log*" >> ~/logfile &
[1] 2492
[1]+  Завершён      find /etc -maxdepth 1 -name "log*" >> ~/logfile
[aeaskerov@fedora ~]$
```

Рис. 4.7: Запуск фонового процесса

7. Удалим файл ~/logfile.

```
[aeaskerov@fedora ~]$ rm logfile
```

Рис. 4.8: Удаление файла logfile

8. Запустим из консоли в фоновом режиме редактор gedit.

```
[aeaskerov@fedora ~]$ gedit &  
[1] 2502
```

Рис. 4.9: Фоновый запуск gedit

9. Определим идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep.

```
[aeaskerov@fedora ~]$ ps | grep "gedit"  
2502 pts/0    00:00:00 gedit  
[aeaskerov@fedora ~]$ ps  
  PID TTY          TIME CMD  
 2249 pts/0    00:00:00 bash  
 2502 pts/0    00:00:00 gedit  
 2549 pts/0    00:00:00 ps  
[aeaskerov@fedora ~]$
```

Рис. 4.10: Определение идентификатора процесса gedit (двумя способами)

10. Прочтём справку (man) команды kill, после чего используем её для завершения процесса gedit.

```
[aeaskerov@fedora ~]$ man kill  
[aeaskerov@fedora ~]$ kill 2502  
[1]+  Завершено      gedit  
[aeaskerov@fedora ~]$
```

Рис. 4.11: Использование команды kill

11. Выполним команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man`.

```
[aeaskerov@fedora ~]$ man df
[aeaskerov@fedora ~]$ df
Файловая система 1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs           4096             0      4096             0% /dev
tmpfs              2006724          0    2006724             0% /dev/shm
tmpfs              802692          1384    801308             1% /run
/dev/sda2          82836480        22490420  57824252            29% /
```

Рис. 4.12: Команда `df`

```
[aeaskerov@fedora ~]$ man du
[aeaskerov@fedora ~]$ du
8      ../mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
8      ../mozilla/extensions
0      ../mozilla/plugins
0      ../mozilla/firefox/Crash Reports/events
```

Рис. 4.13: Команда `du`

12. Воспользовавшись справкой команды `find`, выведем имена всех директорий, имеющих в домашнем каталоге.

```
[aeaskerov@fedora ~]$ man find
[aeaskerov@fedora ~]$ find -maxdepth 1 -type d
.
../mozilla
../.cache
../.config
../.local
../Рабочий стол
```

Рис. 4.14: Имена директорий из домашнего каталога

5 Контрольные вопросы

1. Какие потоки ввода вывода вы знаете?

Ввод и вывод распределяются между тремя стандартными потоками:

- `stdin` — стандартный ввод (клавиатура)
- `stdout` — стандартный вывод (экран)
- `stderr` — стандартная ошибка (вывод ошибок на экран)

2. Объясните разницу между операциями `>` и `»`.

`>` перезаписывает существующий файл или создаёт его, если файл с указанным именем отсутствует в каталоге. `»` добавляет существующий файл или создаёт его, если файл с указанным именем отсутствует в каталоге.

3. Что такое конвейер?

Конвейер (англ. *pipeline*) в терминологии операционных систем семейства Unix — некоторое множество процессов, для которых выполнено следующее перенаправление ввода-вывода: то, что выводит на поток стандартного вывода предыдущий процесс, попадает в поток стандартного ввода следующего процесса.

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

Процесс - это: - программа на стадии выполнения - “объект”, которому выделено процессорное время - асинхронная работа

5. Что такое PID и GID?

PID означает Process Identifier и представляет собой уникальный номер, присвоенный запущенному процессу в операционной системе. Это помогает операционной системе идентифицировать процессы, запущенные в системе, и управлять ими.

GID означает Group Identifier и представляет собой уникальный номер, присвоенный группе пользователей в операционной системе. Это помогает операционной системе управлять контролем доступа и разрешениями для группы пользователей.

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

Задачи — это единицы работы, которые необходимо выполнить в системе Linux. Они могут варьироваться от простых задач, таких как копирование файлов, до сложных задач, таких как управление веб-сервером.

Команда, позволяющая управлять задачами в Linux, называется «ps». Команда «ps» означает «process status» и отображает информацию о запущенных в данный момент процессах в системе. С помощью команды «ps» вы можете просмотреть идентификатор процесса (PID), использование ЦП и памяти, а также другие сведения о каждом процессе. Вы также можете использовать другие команды, такие как «kill», чтобы остановить или завершить процесс, если это необходимо.

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

Утилиты top и htop используются для мониторинга производительности системы и отображения в реальном времени информации о процессах, запущенных в системе. Некоторые из их функций включают в себя:

- Отображение использования системных ресурсов: обе утилиты отображают информацию об использовании ЦП, памяти и активности диска.

- Сортировка процессов. Пользователи могут сортировать процессы по различным критериям, таким как использование ЦП, использование памяти, идентификатор процесса и т. д.
 - Уничтожение процессов. Пользователи могут завершать процессы, вызывающие проблемы или потребляющие слишком много ресурсов.
 - Отображение сведений о процессе: обе утилиты предоставляют подробную информацию о каждом процессе, включая его статус, владельца, приоритет и другие атрибуты.
 - Настройка: пользователи могут настраивать отображение информации, выбирая отображаемые столбцы, изменяя цветовую схему и т. д.
8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.

Команда поиска «find» используется для поиска файлов и каталогов на основе различных критериев поиска, таких как имя, размер, тип и дата модификации. Это мощный инструмент для поиска определенных файлов или групп файлов в иерархии каталогов.

Синтаксис команды следующий:

`find [путь] [аргумент]`

Где «путь» указывает начальный каталог для поиска, а «аргумент» указывает критерии поиска.

Некоторые распространенные варианты включают в себя:

- `-name`: Поиск файлов с определенным именем.
- `-type`: поиск файлов определенного типа (например, каталог, файл, символическая ссылка).
- `-size`: Поиск файлов определенного размера.
- `-mtime`: поиск файлов, измененных в течение определенного периода времени.

Примеры использования команды «find»:

1. Чтобы найти все файлы с определенным именем в текущем каталоге и его подкаталогах:

```
find . -name "*.txt"
```

Это отобразит все файлы с расширением «.txt» в текущем каталоге и его подкаталогах.

2. Чтобы найти все каталоги по определенному пути:

```
find /home/user -type d
```

Это отобразит все каталоги в разделе «/home/user».

3. Чтобы найти все файлы, измененные за последние 7 дней:

```
find . -type f -mtime -7
```

Это отобразит все файлы, измененные за последние 7 дней в текущем каталоге и его подкаталогах.

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

Для поиска файла по содержимому можно воспользоваться командой `grep`.

Пример: `grep -r строка_поиска каталог`

10. Как определить объём свободной памяти на жёстком диске?

Чтобы определить количество свободной памяти на жестком диске, можно использовать следующую команду в терминале: `df -h`. Эта команда отобразит информацию об использовании диска, включая общий размер, используемое пространство и доступное пространство для всех смонтированных файловых систем.

11. Как определить объём вашего домашнего каталога?

Чтобы определить размер вашего домашнего каталога, вы можете использовать следующую команду в терминале: `du -sh ~`. Эта команда отобразит размер вашего домашнего каталога, включая общий размер всех файлов и каталогов в вашем домашнем каталоге.

12. Как удалить зависший процесс?

Чтобы удалить зависший процесс в Linux, вы можете использовать следующие шаги:

1. Определите идентификатор процесса (PID) зависшего процесса с помощью команды `ps aux | grep <имя процесса>`.
2. Используйте команду `kill <PID>`, чтобы завершить процесс.
3. Если процесс все еще не завершен, используйте команду `kill -9 <PID>`, чтобы принудительно завершить процесс.

6 Выводы

Изучены инструменты поиска файлов и фильтрации текстовых данных. Приобретены практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.