

Отчёт по лабораторной работе №6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Гайсина Алина Ринатовна

Содержание

Цель работы	5
Задания	6
Ход работы	7
Вывод	10
Контрольные вопросы.	11

Список иллюстраций

Список таблиц

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.

Задания

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, 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.

Ход работы

1. Вошла в систему.

2. Поднялась в корневой каталог и, воспользовавшись командой `ls` и перенаправлением вывода, записала в `file.txt` названия всех файлов, содержащихся в каталоге `/etc`. Затем я дописала в этом же файле названия файлов, содержащихся в моем домашнем каталоге. (рис. [-@fig:001])

```
[root@fedora /]# ls etc/ > file.txt
```

3. С помощью команды `grep` вывела имена всех файлов из `file.txt`, имеющих расширение `.conf`. (рис. [-@fig:002])

```
[root@fedora /]# grep .conf file.txt
```

Далее с помощью той же команды и перенаправления вывода я записала имена файлов в `conf.txt`. (рис. [-@fig:003])

```
[root@fedora /]# grep .conf file.txt > conf.txt
```

4. Перешла в домашний каталог и с помощью команды `find` вывел названия всех

```
[argayjsina@fedora ~]$ find -name "с*" -pr  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
./mozilla/firefox/0ov6r25f.default-releas  
/caches.sqlite  
./mozilla/firefox/0ov6r25f.default-releas  
com/cache  
./mozilla/firefox/0ov6r25f.default-releas  
com/cache/caches.sqlite
```

файлов, начинавшиеся с символа `с`. (рис. [-@fig:004])

Затем я сделала то же самое с помощью `ls`, `конвейера`, и `grep`. (рис. [-@fig:005])

```
[argayjsina@fedora ~]$ ls -R | grep c*
```

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

```
[root@fedora ~]# find /etc -name "h*" -print
/etc/avahi/hosts
/etc/brltty/Contraction/ha.ctb
/etc/brltty/Input/bm/horizontal.kti
/etc/brltty/Input/hd
/etc/brltty/Input/hm
/etc/brltty/Input/ht
/etc/brltty/Input/hw
/etc/brltty/Text/he.ttb
/etc/brltty/Text/hi.ttb
/etc/brltty/Text/hr.ttb
/etc/brltty/Text/hu.ttb
/etc/brltty/Text/hy.ttb
```

символа h, используя грег.(рис. [-@fig:006])

6. В фоновом режиме запустила команду find и перенаправление вывода, которые будут записывать в файл ~/logfile файлы, имена которых начинаются с log.(рис. [-@fig:007])

```
[root@fedora ~]# find -name "log*" > logfile
```

7. Удалила logfile .(рис. [-@fig:008])

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

8. В фоновом режиме запустила gedit.(рис. [-@fig:009])

```
[argayjsina@fedora ~]$ gedit
```

9. Определила идентификатор процесса gedit, используя команду ps, конвейер и фильтр грег.(рис. [-@fig:010])

```
[argayjsina@fedora ~]$ ps aux | grep gedit
argayjs+  2211  2.6  3.3 787524 67896 pts/0    Sl   20:11   0:01 gedit
argayjs+  2251  1.0  0.1 221812 2328 pts/0    S+   20:11   0:00 grep --color=auto gedit
```

10. Прочитала справку команды kill. Далее, воспользовавшись ею, завершила про-

```
NAME
kill - terminate a process

SYNOPSIS
kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--] pid|name...

kill -l [number] | -L

DESCRIPTION
The command kill sends the specified signal to the specified processes or process groups.

If no signal is specified, the TERM signal is sent. The default action for this signal is to terminate the process. This signal should be used in preference to the KILL signal (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a TERM signal has been sent, then the KILL signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.

Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.

If signal is 0, then no actual signal is sent, but error checking is still performed.

ARGUMENTS
The list of processes to be signaled can be a mixture of names and PIDs.

pid
Each pid can be expressed in one of the following ways:

n
where n is larger than 0. The process with PID n is signaled.

0
All processes in the current process group are signaled.

-1
```

цесс gedit.(рис. [-@fig:011;-@fig:012])


```
[argayjsina@fedora ~]$ kill 2211
[1]+  Завершено      gedit
```

11. Предварительно получив более подробную информацию о `df` и `du`, с помощью

```
[argayjsina@fedora ~]$ man df
[argayjsina@fedora ~]$ df -vi
Файловая система  Инодов  ИИспользовано  ИСвободно  ИИспользовано%  Смонтиро
devtmpfs          246948      461    246487          1% /dev
tmpfs             252044        1    252043          1% /dev/shm
tmpfs             819200      878    818322          1% /run
/dev/sda2          0            0         0          - /
tmpfs             409600        46    409554          1% /tmp
/dev/sda2          0            0         0          - /home
/dev/sda1          65536      450     65086          1% /boot
tmpfs             50408      122     50286          1% /run/usi
[argayjsina@fedora ~]$ man du
[argayjsina@fedora ~]$ du -a
```

команды `man`, воспользовалась ими. (рис. [-@fig:013])

12. С помощью команды `find` вывела название директорий, находящихся в домаш-

```
[argayjsina@fedora ~]$ find -type d
```

нем каталоге. (рис. [-@fig:014])

Вывод

Мы научились пользоваться инструментами поиска файлов и фильтрации текстовых данных. Приобрели практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

Контрольные вопросы.

1. Какие потоки ввода вывода вы знаете?

- 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. Объясните разницу между операцией `>` и `>>`.

- `>` - перенаправление вывода (`stdout`) в файл.
- `>>` - Перенаправление вывода (`stdout`) в файл, но при этом он открывается в режиме добавления.

3. Что такое конвейер?

- Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.

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

- 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. Что такое PID и GID?

- Каждому процессу Linux или Unix или выполняемой программе, автоматически присваивается идентификационный номер уникального процесса (PID). PID автоматически присваивает номер для каждого процесса в системе.
- Кроме идентификационного номера пользователя с учётной записью связан идентификатор группы. Группы пользователей применяются для организации доступа нескольких пользователей к некоторым ресурсам. У группы, так же, как и у пользователя, есть имя и идентификационный номер — GID

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

- Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач. Для завершения задачи необходимо выполнить команду kill.

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

- top - интерактивный просмотрщик процессов. htop аналог top.

9. Назовите и дайте характеристику команде поиска файлов.

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

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

- grep -r -n «text» /path , где -n показывает строку, где был найден фрагмент, а -r осуществляет розыск рекурсивно, в файлах в самом каталоге /path и в его подкаталогах;

11. Как определить объем свободной памяти на жёстком диске?

- С помощью команды df -h можно посмотреть объем занятой и свободной памяти на жестком диске.

12. Как определить объем вашего домашнего каталога?

- Воспользоваться командой, указанной выше, и постараться найти домашний каталог среди всех остальных.

13. Как удалить зависший процесс?

- Узнать его идентификационный номер и воспользоваться командой `kill`.