

Отчёт по лабораторной работе №6

Дисциплина: Операционные системы

Елизавета Андреевна Алмазова

Содержание

Цель работы	1
Задание	1
Теоретическое введение	2
Потоки ввода-вывода	2
Конвейер	2
Работа с файлами и текстами	2
Использование диска	3
Управление задачами и процессами	3
Выполнение лабораторной работы	3
Выводы	6
Ответы на контрольные вопросы	6

Цель работы

Цель данной лабораторной 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.

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, начинавшиеся с символа с? 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, имеющих в вашем домашнем каталоге.

Теоретическое введение

Потоки ввода-вывода

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

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

Работа с файлами и текстами

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

Найти в текстовом файле указанную строку символов позволяет команда `grep`. Формат команды: `grep строка имя_файла`. Кроме того, команда `grep` способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом `grep`.

Использование диска

Команда `df` показывает размер каждого смонтированного раздела диска. Формат команды: `df [-опции] [файловая_система]`, команда `du` показывает число килобайт, используемое каждым файлом или каталогом. Формат команды: `du [-опции] [имя_файла...]`. На `afs` можно посмотреть использованное пространство командой `fs quota`

Управление задачами и процессами

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

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. Для получения информации о процессах, управляемых вами и запущенных (работающих или остановленных) на вашем терминале, используйте опцию `aux`.

Выполнение лабораторной работы

1. Осуществила вход в систему, используя пользователя `eaalmazova`.
2. Записала в файл `file.txt` названия файлов, содержащихся в каталоге `/etc` (команда `ls -a /etc > file.txt`). Дописала в этот же файл названия файлов, содержащихся в моем домашнем каталоге (`ls -a ~ >> file.txt`). Командой `grep -e '.conf' 'file.txt'` вывела в консоль имена всех файлов из `file.txt`, имеющих расширение `.conf`, после чего выполнила команду `grep -e '.conf' file.txt > conf.txt` (рис. 1).

```

[eaalmazova@fedora ~]$ grep -e '\.conf$' file.txt
anthy-unicode.conf
appstream.conf
asound.conf
brltty.conf
chrony.conf
dleyna-renderer-service.conf
dleyna-server-service.conf
dnsmasq.conf
dracut.conf
extlinux.conf
fprintd.conf
fuse.conf
host.conf
idmapd.conf
jwhois.conf
kdump.conf
krb5.conf
ld.so.conf
libaudit.conf
libuser.conf
locale.conf
logrotate.conf
man_db.conf
mke2fs.conf
mttools.conf
nfs.conf
nfsmount.conf
nsswitch.conf
opensc.conf
opensc-x86_64.conf
passwdqc.conf
radvd.conf
request-key.conf
resolv.conf
rsyncd.conf
rygel.conf
sestatus.conf
sudo.conf
swtpm-localca.conf
swtpm_setup.conf
sysctl.conf
tcsd.conf
Trolltech.conf
updatedb.conf
uresourced.conf
usb_modeswitch.conf
vconsole.conf
xattr.conf
[eaalmazova@fedora ~]$ grep -e '\.conf$' file.txt > conf.txt

```

Рисунок 1 - Вывод в консоль имен из file.txt, имеющих расширение .conf

3. Определила, какие файлы в моем домашнем каталоге имеют имена, начинающиеся с символа c: conf.txt. Это можно сделать с помощью 3 команд: find ~ -maxdepth 1 -name "C" -print, ls -a ~/c, ls -a ~ | grep c* (рис. 2)

```

[eaalmazova@fedora ~]$ find ~ -maxdepth 1 -name "c*" -print
/home/eaalmazova/conf.txt
[eaalmazova@fedora ~]$ ls -a ~/c*
/home/eaalmazova/conf.txt
[eaalmazova@fedora ~]$ ls -a ~ | grep c*
conf.txt

```

Рисунок 2 - Поиск в домашнем каталоге файлов, начинающихся с символа c

4. Вывела на экран по странично имена файлов из каталога /etc, начинающиеся с символа h, с помощью команды ls -a /etc | grep h* | less. Запустила в фоновом режиме процесс, который записывает в файл ~/logfile файлы, имена которых начинаются с log. Для этого я зашла через суперпользователя и воспользовалась командой ls -aR / | grep log* > ~/logfile &. Удалила файл через rm.
5. Запустила из консоли в фоновом режиме редактор gedit: gedit &. Определила идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep (ps aux | grep gedit, рис.3). Также это можно сделать через команды pidof gedit и pgrep gedit (рис.4).

```

[root@fedora eaalmazova]# ps aux | grep gedit

```

Рисунок 3 - ps aux | grep gedit

```
[root@fedora eaalmazova]# pidof gedit
2605
[root@fedora eaalmazova]# pgrep gedit
2605
```

Рисунок 4 - Другие способы нахождения PID

6. Прочитала справку man команды kill, после чего использовала её для завершения процесса gedit: kill 2605 (рис.5).

```
KILL(1)                                User Commands                                KILL(1)

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
            All processes with a PID larger than 1 are signaled.

        -n
            where n is larger than 1. All processes in process group n are signaled. When an argument of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '--' option, otherwise it will be taken as the signal to send.

    name
        All processes invoked using this name will be signaled.

OPTIONS
    -s, --signal signal
        The signal to send. It may be given as a name or a number.

    -l, --list [number]
        Print a list of signal names, or convert the given signal number to a name. The signals can be found in /usr/include/linux/signal.h.
```

Рисунок 5 - man kill

7. Выполнила команды df и du, предварительно получив более подробную информацию об этих командах, с помощью команды man (рис.6).

```
[root@fedora eaalmazova]# df
Filesystem      1K-blocks      Used Available Use% Mounted on
devtmpfs         1983800          0    1983800   0% /dev
tmpfs            2004136          0    2004136   0% /dev/shm
tmpfs            801656        1376     800280   1% /run
/dev/sda2       103808000    6574560    95543184   7% /
tmpfs            2004140          68     2004072   1% /tmp
/dev/sda2       103808000    6574560    95543184   7% /home
/dev/sda1         996780       175204     752764  19% /boot
OC              976744444    116182944    860561500  12% /media/sf_
tmpfs            400824        120     400704   1% /run/user/1000
```

Рисунок 6 - Выполнение команды df

8. Воспользовавшись справкой команды find, вывела имена всех директорий, имеющих в моем домашнем каталоге и его подкаталогах (find /home/eaalmazova -type d), а затем только директорий в домашнем каталоге: find /home/eaalmazova -type d -maxdepth 1 (рис.7).

```
/home/eaalmazova
/home/eaalmazova/.mozilla
/home/eaalmazova/.cache
/home/eaalmazova/.config
/home/eaalmazova/.local
/home/eaalmazova/Desktop
/home/eaalmazova/Downloads
/home/eaalmazova/Templates
/home/eaalmazova/Public
/home/eaalmazova/Documents
/home/eaalmazova/Music
/home/eaalmazova/Pictures
/home/eaalmazova/Videos
/home/eaalmazova/.gnupg
/home/eaalmazova/work
/home/eaalmazova/.ssh
/home/eaalmazova/.texlive2021
/home/eaalmazova/bin
/home/eaalmazova/reports
/home/eaalmazova/monthly
/home/eaalmazova/ski.places
/home/eaalmazova/australia
/home/eaalmazova/play
```

Рисунок 7 - Директории, имеющиеся в домашнем каталоге без учета его подкаталогов

Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с инструментами поиска файлов и фильтрации текстовых данных, приобрела практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

Ответы на 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 `>>`.

Символ `>` перезаписывает информацию из файла, если там уже что-то есть. Для добавления данных в конец используется `>>`.

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. Конвейер служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий: команда 1 | команда 2. Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например: `1 ls -la | sort > sortilg_list`

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

Процесс - это выполнение программы. Он считается активной сущностью и реализует действия, указанные в программе.

Основное различие между программой и процессом заключается в том, что программа представляет собой группу инструкций для выполнения определенной задачи, тогда как процесс представляет собой программу в процессе выполнения. Между процессом и программой существует отношение многие-к-одному, что означает, что одна программа может вызывать несколько процессов или, другими словами, несколько процессов могут быть частью одной и той же программы.

5. Что такое PID и GID?

Любой команде, выполняемой в системе, присваивается идентификатор процесса (process ID, PID). GID — идентификационный номер основной группы пользователя.

6. 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.

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

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. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.

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

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Примеры: `find ~ -name "f" -print` выводит на экран имена файлов из домашнего каталога и его подкаталогов, начинающихся на `f`, `find /etc -name "p" -print` выводит на экран имена файлов в каталоге `/etc`, начинающихся с символа `p`.

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

Можно. С помощью команды `grep -iRl "фраза" /директория/где/искать`

Например, команда может выглядеть вот так: `grep -iRl "sun" /home`

Команда найдет и выведет все файлы, которые содержат фразу `sun` в директории `/home` и во всех директориях, внутри этой папки. Используются следующие ключи:

- `i` - игнорировать регистра текста (большие или маленькие буквы);
- `R` - рекурсивно искать файлы в субдиректориях;
- `I` - показывать названия файлов, вместо их содержимого;

10. Как определить объем свободной памяти на жёстком диске?

С помощью команды `df`

11. Как определить объем вашего домашнего каталога?

С помощью команды `du /home/eaalmazova`

12. Как удалить зависший процесс?

Для удаления сначала потребуется определить идентификатор процесса `PID`. Команда `pidof [процесс]` выведет требуемый `PID`. Послать сигнал процессу с помощью команды `kill`: `kill` пошлет сигнал о завершении процесса, однако этот сигнал можно перехватить или заблокировать, а `kill -9` посылает сигнал об уничтожении процесса без возможности перехвата или игнорирования. Есть и другие способы.