

Отчёт по лабораторной работе №6

Дисциплина: операционные системы

Лебедева Алёна Алексеевна

Содержание

Цель работы	1
Задание	1
Выполнение лабораторной работы.....	2
Контрольные вопросы	7
Выводы	9

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.

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 с? Предложите несколько вариантов, как это сделать.
5. Выведите на экран (постранично) 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.

Выполнение лабораторной работы

1. Захожу в систему с помощью логина и пароля
2. Для того, чтобы записать в файл file.txt названия файлов, содержащихся в каталоге /etc, использую команду «ls /etc > file.txt». Далее с помощью команды «ls ~ >> file.txt» дописываю в этот же файл названия файлов, содержащихся в моем домашнем каталоге.

Вывожу имена всех файлов из file.txt, имеющих расширение .conf и записываю их в новый текстовый файл conf.txt с помощью команды «grep .conf file.txt (рис. [-@fig:001])

```
[aalebedeva@fedora ~]$ ls /etc > file.txt
[aalebedeva@fedora ~]$ ls ~ >>file.txt
[aalebedeva@fedora ~]$ grep .conf file.txt
anthy-unicode.conf
appstream.conf
asound.conf
brltty.conf
chkconfig.d
chrony.conf
dconf
dleyna-renderer-service.conf
dleyna-server-service.conf
dnsmasq.conf
dracut.conf
dracut.conf.d
extlinux.conf
fprintd.conf
fuse.conf
host.conf
idmapd.conf
jwhois.conf
kdump.conf
krb5.conf
krb5.conf.d
ld.so.conf
ld.so.conf.d
libaudit.conf
libuser.conf
locale.conf
logrotate.conf
makedumpfile.conf.sample
man_db.conf
mke2fs.conf
mtools.conf
```

запись в файлы

3. Чтобы вывести на экран файлы, имена которых начинаются с символа c, я использую команды `ls -l | grep c`; `find ~/c -name "c*" -print`

(рис. [-@fig:002])

```
[aalebedeva@fedora ~]$ grep .conf file.txt > conf.txt
[aalebedeva@fedora ~]$ ls -l | grep c*
-rw-rw-r--. 1 aalebedeva aalebedeva 805 мая 6 11:35 conf.txt
[aalebedeva@fedora ~]$ find ~/c* -name "*" -print
/home/aalebedeva/conf.txt
[aalebedeva@fedora ~]$ ls -l | grep c*
```

вывод файлов

4. Чтобы вывести на экран (постранично) имена файлов из каталога /etc, начинающиеся с символа h, воспользуемся командой «`find /etc -name "h*" -print`»

(рис. [-@fig:003])

```
[aalebedeva@fedora etc]$ find /etc -name "h*" -print
find: '/etc/audit': Отказано в доступе
/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
/etc/containers/oci/hooks.d
find: '/etc/cups/ssl': Отказано в доступе
find: '/etc/dhcp': Отказано в доступе
find: '/etc/firewalld': Отказано в доступе
find: '/etc/grub.d': Отказано в доступе
/etc/hp
/etc/hp/hplip.conf
/etc/httpd
/etc/httpd/conf/httpd.conf
/etc/libibverbs.d/hfiverbs.driver
/etc/libibverbs.d/hns.driver
find: '/etc/libvirt': Отказано в доступе
/etc/logrotate.d/httpd
find: '/etc/lvm/archive': Отказано в доступе
find: '/etc/lvm/backup': Отказано в доступе
find: '/etc/lvm/cache': Отказано в доступе
find: '/etc/nftables': Отказано в доступе
```

вывод файлов

5. Запускаю в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log, используя команду «find ~ -name "log*" -print > ~/logfile &» Удаляю файл ~/logfile командой «rm logfile»

(рис. [-@fig:004])

```
[aalebedeva@fedora etc]$ find ~ -name "log*" -print > ~/logfile &
[1] 5515
[aalebedeva@fedora etc]$ rm -r logfile
```

logfile

6. Запускаю из консоли в фоновом режиме редактор gedit.

(рис. [-@fig:005])

```
[aalebedeva@fedora etc]$ gedit &  
[2] 5579  
[1] Выход 127      gedot  
[aalebedeva@fedora etc]$
```

gedit

7. Чтобы определить идентификатор процесса `gedit`, использую команду «`ps | grep -i "gedit"`». Наш процесс имеет PID 518. Узнать идентификатор процесса можно также, используя команду «`pgrep gedit`» или «`pidof gedit`»
8. Прочитав информацию о команде `kill` с помощью команды «`man kill`», использую её для завершения процесса `gedit`

(рис. [-@fig:006])

```
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.  
  
Manual page kill(1) line 1 (press h for help or q to quit)
```

kill

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

(рис. [-@fig:007])

```
[aalebedeva@fedora etc]$ df
Файловая система 1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs          2451208           0  2451208           0% /dev
tmpfs             2471592           0  2471592           0% /dev/shm
tmpfs             988640          1376  987264           1% /run
/dev/sda2         82835456       11219936  70571840          14% /
tmpfs             2471596           76  2471520           1% /tmp
/dev/sda2         82835456       11219936  70571840          14% /home
/dev/sda1         996780          175300   752668           19% /boot
tmpfs             494316           124  494192           1% /run/user/1000
/dev/sr0          60096          60096           0         100% /run/media/aalebe
deva/VBox_GAs_6.1.34
[aalebedeva@fedora etc]$ du
0      ./java/.systemPrefs
0      ./java
0      ./NetworkManager/conf.d
0      ./NetworkManager/dispatcher.d/no-wait.d
0      ./NetworkManager/dispatcher.d/pre-down.d
0      ./NetworkManager/dispatcher.d/pre-up.d
0      ./NetworkManager/dispatcher.d
0      ./NetworkManager/dnsmasq-shared.d
0      ./NetworkManager/dnsmasq.d
0      ./NetworkManager/system-connections
4      ./NetworkManager
12     ./PackageKit
4      ./UPower
```

df u du

10. Получаем информацию с помощью команды «`man find`» и выводим имена всех директорий, имеющихя в домашнем каталоге с помощью команды «`find ~ - type d -print`»

(рис. [-@fig:008])


```
[aalebedeva@fedora etc]$ find ~ -type d -print
/home/aalebedeva
/home/aalebedeva/.mozilla
/home/aalebedeva/.mozilla/extensions
/home/aalebedeva/.mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
/home/aalebedeva/.mozilla/plugins
/home/aalebedeva/.mozilla/firefox
/home/aalebedeva/.mozilla/firefox/Crash Reports
/home/aalebedeva/.mozilla/firefox/Crash Reports/events
/home/aalebedeva/.mozilla/firefox/Pending Pings
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/minidumps
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/crashes
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/crashes/events
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/security_state
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent/chrome
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent/chrome/idb
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-es.files/journals
/home/aalebedeva/.mozilla/firefox/9ikuutj9.default-release/storage/permanent/chrome/idb/3561288849sdhlie.files
```

find

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. Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода `stdout`.
2. “>” Перенаправление вывода в файл “>>” Перенаправление вывода в файл и открытие файла в режиме
3. Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.
4. Процесс рассматривается операционной системой как заявка на потребление всех видов ресурсов, кроме одного – процессорного времени. Этот последний важнейший ресурс распределяется операционной системой между другими единицами работы – потоками, которые и получили свое название благодаря тому, что они представляют собой последовательности (потоки выполнения)

команд. Процесс – это выполнение программы. Он считается активной сущностью и реализует действия, указанные в программе. Программа представляет собой статический набор команд, а процесс – это набор ресурсов и данных, использующихся при выполнении программы.

5. `pid`: идентификатор процесса (PID) процесса (process ID), к которому вызывают метод `gid`: идентификатор группы UNIX, в котором работает программа.
6. `top` – это консольная программа, которая показывает список работающих процессов в системе. Программа в реальном времени отсортирует запущенные процессы по их нагрузке на процессор. `htop` – это продвинутый консольный мониторинг процессов. Утилита выводит постоянно меняющийся список системных процессов, который сортируется в зависимости от нагрузки на ЦПУ. Если делать сравнение с `top`, то `htop` показывает абсолютно все процессы в системе, время их непрерывного использования, загрузку процессоров и расход оперативной памяти.
7. 8. `find` – 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.

Команда `find` имеет такой синтаксис: `find папка параметры критерий шаблон действие`
Папка – каталог в котором будем искать. Параметры – дополнительные параметры, например, глубина поиска, и т.д. Критерий – по какому критерию будем искать: имя, дата создания, права, владелец и т.д. Шаблон – непосредственно значение по которому будем отбирать файлы.
Основные параметры: • `-P` никогда не открывать символические ссылки • `-L` – получает информацию о файлах по символическим ссылкам. Важно для дальнейшей обработки, чтобы обрабатывалась не ссылка, а сам файл • `-maxdepth` – максимальная глубина поиска по подкаталогам, для поиска только в текущем каталоге установите 1 • `-depth` – искать сначала в текущем каталоге, а потом в подкаталогах • `-mount` искать файлы только в этой файловой системе • `-version` – показать версию утилиты `find` • `-print` – выводить полные имена файлов • `-type f` – искать только файлы • `-type d` – поиск папки в Linux
Основные критерии: • `-name` – поиск файлов по имени • `-perm` – поиск файлов в Linux по режиму доступа • `-user` – поиск файлов по владельцу • `-group` – поиск по группе • `-mtime` – поиск по времени модификации файла • `-atime` – поиск файлов по дате последнего чтения • `-nogroup` – поиск файлов, не принадлежащих ни одной группе • `-nouser` – поиск файлов без владельцев • `-newer` – найти файлы новее чем указанный • `-size` – поиск файлов в Linux по их размеру
Примеры: `find ~ -type d` поиск директорий в домашнем каталоге `find ~ -type f -name “.*”` поиск скрытых файлов в домашнем каталоге

8. Файл по его содержимому можно найти с помощью команды `grep`: «`grep -r “слово/выражение, которое нужно найти”`».

9. Утилита `df`, позволяет проанализировать свободное пространство на всех подключенных к системе разделах.
10. Для домашнего каталога: `du ~/`
11. Основные сигналы (каждый сигнал имеет свой номер), которые используются для завершения процесса: • `SIGINT` – самый безобидный сигнал завершения, означает `Interrupt`. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш `Ctrl+C`. Процесс правильно завершает все свои действия и возвращает управление; • `SIGQUIT` – это еще один сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей, что нужно завершиться и программа может выполнить корректное завершение или проигнорировать сигнал. В отличие от предыдущего, она генерирует дампы памяти. Сочетание клавиш `Ctrl+;`; • `SIGHUP` – сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом; • `SIGTERM` – немедленно завершает процесс, но обрабатывается программой, поэтому позволяет ей завершить дочерние процессы и освободить все ресурсы; • `SIGKILL` – тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными. Также для передачи сигналов процессам в Linux используется утилита `kill`, её синтаксис: `kill -сигнал pid_процесса` (`PID` – уникальный идентификатор процесса). Сигнал представляет собой один из выше перечисленных сигналов для завершения процесса. Перед тем, как выполнить остановку процесса, нужно определить его `PID`. Для этого используют команды `ps` и `grep`. Команда `ps` предназначена для вывода списка активных процессов в системе и информации о них. Команда `grep` запускается одновременно с `ps` (в канале) и будет выполнять поиск по результатам команды `ps`. Утилита `pkill` – это оболочка для `kill`, она ведет себя точно так же, и имеет тот же синтаксис, только в качестве идентификатора процесса ей нужно передать его имя. `killall` работает аналогично двум предыдущим утилитам. Она тоже принимает имя процесса в качестве параметра и ищет его `PID` в директории `/proc`. Но эта утилита обнаружит все процессы с таким именем и завершит их.

Выводы

Ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.