

Лабораторная работа №8

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Дагделен Зейнап Реджеповна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	15
6	Ответы на контрольные вопросы	16

Список иллюстраций

4.1	Запись в файл	8
4.2	Вывод содержимого файла	8
4.3	Добавление данных в файл	9
4.4	Поиск файлов определенного расширения	9
4.5	Запись в файл	9
4.6	Поиск файлов, начинающихся с определенного элемента	10
4.7	Поиск файлов, начинающихся с определенного элемента	10
4.8	Поиск файлов, начинающихся с определенного элемента	10
4.9	Создание фонового процесса	10
4.10	Удаление файла	11
4.11	Создание фонового процесса	11
4.12	Поиск идентификатора процесса	11
4.13	Чтение документации	12
4.14	Удаление процесса	12
4.15	Чтение документации	12
4.16	Утилита df	13
4.17	Утилита du	13
4.18	Чтение документации	13
4.19	Название рисунка	14

Список таблиц

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 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 директорий, имеющихся в вашем домашнем каталоге.

3 Теоретическое введение

В интерфейсе командной строки есть очень полезная возможность перенаправления (переадресации) ввода и вывода (англ. термин I/O Redirection). Как мы уже заметили, многие программы выводят данные на экран. А ввод данных в терминале осуществляется с клавиатуры. С помощью специальных обозначений можно перенаправить вывод многих команд в файлы или иные устройства вывода (например, распечатать на принтере). Тоже самое и со вводом информации, вместо ввода данных с клавиатуры, для многих программ можно задать считывание символов их файла. Кроме того, можно даже вывод одной программы передать на ввод другой программе.

К каждой программе, запускаемой в командной строке, по умолчанию подключено три потока данных:

STDIN (0) — стандартный поток ввода (данные, загружаемые в программу). STDOUT (1) — стандартный поток вывода (данные, которые выводит программа). По умолчанию — терминал. STDERR (2) — стандартный поток вывода диагностических и отладочных сообщений (например, сообщениях об ошибках). По умолчанию — терминал.

Pipe (конвейер) – это однонаправленный канал межпроцессного взаимодействия. Термин был придуман Дугласом Макилроем для командной оболочки Unix и назван по аналогии с трубопроводом. Конвейеры чаще всего используются в shell-скриптах для связи нескольких команд путем перенаправления вывода одной команды (stdout) на вход (stdin) последующей, используя символ конвейера '|'.

4 Выполнение лабораторной работы

Я вошла в систему под соответствующим 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. fig. 4.1).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo ls -lR /etc > file.txt
[sudo] пароль для zrdagdelen:
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.1: Запись в файл

Проверила, что в файл записались нужные значения с помощью утилиты head, она выводит первые 10 строк файла на экран (рис. fig. 4.2).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ head file.txt
/etc:
итого 1248
drwxr-xr-x 3 root root 4096 фев 20 22:24 acpi
-rw-r--r-- 1 root root 3028 фев 20 22:22 adduser.conf
drwxr-xr-x 3 root root 4096 фев 20 22:23 alsa
drwxr-xr-x 2 root root 12288 мар 21 19:56 alternatives
-rw-r--r-- 1 root root 335 мар 23 2022 anacrontab
drwxr-xr-x 3 root root 4096 мар 8 15:40 apache2
-rw-r--r-- 1 root root 433 мар 23 2022 apg.conf
drwxr-xr-x 5 root root 4096 фев 20 22:23 apt
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.2: Вывод содержимого файла

Добавила в созданный файл имена файлов из домашнего каталога, используя перенаправление “>” в режиме добавления (рис. fig. 4.3).


```
zrdagdelen@zrdagdelen-MacBookPro:~$ ls -lR ~/ >> file.txt
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.3: Добавление данных в файл

Вывела на экран имена всех файлов, имеющих расширение “.conf” с помощью утилиты grep (рис. fig. 4.4).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ grep .conf file.txt
-rw-r--r-- 1 root root 3028 фев 20 22:22 adduser.conf
-rw-r--r-- 1 root root 433 мар 23 2022 apg.conf
-rw-r--r-- 1 root root 769 фев 22 2022 appstream.conf
-rw-r--r-- 1 root root 29219 июн 28 2022 brltty.conf
-rw-r--r-- 1 root root 6253 фев 20 22:22 ca-certificates.conf
-rw-r--r-- 1 root root 5529 фев 20 22:22 ca-certificates.conf.dpkg-old
drwxr-xr-x 4 root root 4096 фев 20 22:23 dconf
-rw-r--r-- 1 root root 2969 фев 20 2022 debconf.conf
-rw-r--r-- 1 root root 604 сен 16 2018 deluser.conf
-rw-r--r-- 1 root root 685 янв 8 2022 e2scrub.conf
-rw-r--r-- 1 root root 20 фев 24 2022 fprintd.conf
-rw-r--r-- 1 root root 694 мар 23 2022 fuse.conf
-rw-r--r-- 1 root root 2584 фев 2 2022 gai.conf
```

Рис. 4.4: Поиск файлов определенного расширения

Добавила вывод прошлой команды в новый файл conf.txt с помощью перенаправления “>” (файл создается при выполнении этой команды) (рис. fig. 4.5).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ grep .conf file.txt > conf.txt
zrdagdelen@zrdagdelen-MacBookPro:~$ head conf.txt
-rw-r--r-- 1 root root 3028 фев 20 22:22 adduser.conf
-rw-r--r-- 1 root root 433 мар 23 2022 apg.conf
-rw-r--r-- 1 root root 769 фев 22 2022 appstream.conf
-rw-r--r-- 1 root root 29219 июн 28 2022 brltty.conf
-rw-r--r-- 1 root root 6253 фев 20 22:22 ca-certificates.conf
-rw-r--r-- 1 root root 5529 фев 20 22:22 ca-certificates.conf.dpkg-old
drwxr-xr-x 4 root root 4096 фев 20 22:23 dconf
-rw-r--r-- 1 root root 2969 фев 20 2022 debconf.conf
-rw-r--r-- 1 root root 604 сен 16 2018 deluser.conf
-rw-r--r-- 1 root root 685 янв 8 2022 e2scrub.conf
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.5: Запись в файл

Определяю, какие файлы в домашнем каталоге начинаются с символа “с” 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 кол-во любых символов, добавляю опцию -print, чтобы мне вывелся результат (рис. fig. 4.6). Но таким образом я получаю информацию даже о файлах из подкаталогов домашнего каталога.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ find ~ -name "c*" -print
/home/zrdagdelen/.cache/mesa_shader_cache/68/c8a6795f9fe0b92afdb4423cdbf02febe4e
c98
/home/zrdagdelen/.cache/mesa_shader_cache/c4
/home/zrdagdelen/.cache/mesa_shader_cache/96/c4adf1dd5cef8d3fc47c9fcf77a030aeaac
7ef
/home/zrdagdelen/.cache/mesa_shader_cache/ca
/home/zrdagdelen/.cache/mesa_shader_cache/fc/c0830be4da4815491df546909d0d2e1c5d5
6ba
/home/zrdagdelen/.cache/mesa_shader_cache/86/cae490d0fd93ce2afc7e38ad80e01c6efa1
af5
/home/zrdagdelen/.cache/mesa_shader_cache/c7
/home/zrdagdelen/.cache/mesa_shader_cache/54/cc638be4b9db09176fe7c17004177feff7f
9ff
```

Рис. 4.6: Поиск файлов, начинающихся с определенного элемента

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. fig. 4.7).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ ls -lR | grep c*
-rw-rw-r-- 1 zrdagdelen zrdagdelen 42686 мар 29 19:00 conf.txt
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.7: Поиск файлов, начинающихся с определенного элемента

С помощью метода `find`, чьи опции я расписала ранее, ищу все файлы, начинающиеся с буквы `h` (рис. fig. 4.8).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ sudo find /etc -name "h*" -print
/etc/hosts.allow
/etc/X11/cursors/handhelds.theme
/etc/host.conf
/etc/brltty/Input/hm
/etc/brltty/Input/ht
/etc/brltty/Input/hw
/etc/brltty/Input/bm/horizontal.kti
/etc/brltty/Input/hd
/etc/brltty/Contraction/ha.ctb
/etc/brltty/Text/hc.tth
```

Рис. 4.8: Поиск файлов, начинающихся с определенного элемента

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. fig. 4.9).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ find ~ -name "long*" -print > logfile &
[1] 6692
```

Рис. 4.9: Создание фонового процесса

Проверяю, что файл создан, удаляю его, проверяю, что файл удален (рис. fig. 4.10).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ ls
conf.txt      package.json  snap          Изображения
file.txt      package-lock.json work           Музыка
install-tl-20240308 pandoc-3.1.12.2 Видео        Общедоступные
logfile       pandoc-crossref Документы     'Рабочий стол'
node_modules  pandoc-crossref.1 Загрузки      Шаблоны
zrdagdelen@zrdagdelen-MacBookPro:~$ rm logfile
zrdagdelen@zrdagdelen-MacBookPro:~$ ls
conf.txt      package-lock.json work           Музыка
file.txt      pandoc-3.1.12.2 Видео        Общедоступные
install-tl-20240308 pandoc-crossref Документы     'Рабочий стол'
node_modules  pandoc-crossref.1 Загрузки      Шаблоны
package.json  snap          Изображения
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.10: Удаление файла

Запускаю в консоли в фоновом режиме (с помощью символа &) редактор mousepad (рис. fig. 4.11).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ mousepad &
[1] 9632
```

Рис. 4.11: Создание фонового процесса

С помощью утилиты ps определяю идентификатор процесса mousepad, его значение 9632 (рис. fig. 4.12). Также мы можем определить идентификатор с помощью rgrep.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ ps aux | grep mousepad
zrdagde+  9632  4.1  0.3 901384 60912 pts/0    SL  19:35   0:00 mousepad
zrdagde+  9652  0.0  0.0 17892  2560 pts/0    S+  19:35   0:00 grep --color=
auto mousepad
zrdagdelen@zrdagdelen-MacBookPro:~$ pgrep mousepad
9632
zrdagdelen@zrdagdelen-MacBookPro:~$ ps aux | grep mousepad | grep -v grep
zrdagde+  9632  0.1  0.3 901384 60912 pts/0    SL  19:35   0:00 mousepad
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.12: Поиск идентификатора процесса

Прочитала справку команды kill (рис. fig. 4.13).

```
zrdagdelen@zrdagdelen-MacBookPro: ~  
KILL(1) User Commands KILL(1)  
NAME  
kill - send a signal to a process  
SYNOPSIS  
kill [options] <pid> [...]  
DESCRIPTION  
The default signal for kill is TERM. Use -l or -L to list available  
signals. Particularly useful signals include HUP, INT, KILL, STOP,  
CONT, and 0. Alternate signals may be specified in three ways: -9,  
-SIGKILL or -KILL. Negative PID values may be used to choose whole  
process groups; see the PGID column in ps command output. A PID of -1  
is special; it indicates all processes except the kill process itself  
and init.
```

Рис. 4.13: Чтение документации

Использую команду kill и идентификатор процесса, чтобы его удалить (рис. fig. 4.14). Как и ожидалось, закрылась программа mousepad.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ pgrep mousepad  
9632  
zrdagdelen@zrdagdelen-MacBookPro:~$ ps aux | grep mousepad | grep -v grep  
zrdagde+  9632  0.1  0.3 901384 60912 pts/0    Sl   19:35   0:00 mousepad  
zrdagdelen@zrdagdelen-MacBookPro:~$ man kill  
zrdagdelen@zrdagdelen-MacBookPro:~$ kill 9632  
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.14: Удаление процесса

Прочитала документацию про функции df и du (рис. fig. 4.15).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ man df  
[1]+  Завершён      mousepad  
zrdagdelen@zrdagdelen-MacBookPro:~$ man du  
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.15: Чтение документации

Использую утилиту df опции -iv позволяют увидеть информацию об инодах и сделать вывод читаемым, игнорируя сообщения системы о нем (рис. fig. 4.16). Эта утилита нам нужна, чтобы выяснить, сколько свободного места есть у нашей системы.

```
zrdagdelen@zrdagdelen-MacBookPro:~$ df -vi
Файл.система      Инодов  Ииспользовано  Исвободно  Ииспользовано%  Смонтировано в
tmpfs              2033955      1236         2032719         1% /run
/dev/sdb3          7815168      749501      7065667        10% /
tmpfs              2033955         1         2033954         1% /dev/shm
tmpfs              2033955         4         2033951         1% /run/lock
/dev/sdb1           0             0             0             - /boot/efi
tmpfs              406791       161         406630         1% /run/user/1000
/dev/sdb2          4294967295    142 4294967153    1% /media/zrdagde
len/SSD tasinir
zrdagdelen@zrdagdelen-MacBookPro:~$
```

Рис. 4.16: Утилита df

Используя утилиту du. Она нужна чтобы просмотреть, сколько места занимают файлы в определенной директории и найти самые большие из них (рис. fig. 4.17).

```
zrdagdelen@zrdagdelen-MacBookPro:~$ du -a work/study/2023-2024/'Операционные сис
темы'/os-intro/labs/lab01/
8 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/bib
/cite.bib
12 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/bib
8 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/rep
ort.md
252 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/ima
ge/placeimg_800_600_tech.jpg
256 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/ima
ge
268 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/rep
ort.docx
16 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/pan
doc/csl/gost-r-7-0-5-2008-numeric.csl
20 work/study/2023-2024/Операционные системы/os-intro/labs/lab01/report/pan
doc/csl
```

Рис. 4.17: Утилита du

Прочитала документацию о команде find (рис. fig. 4.18).

```
zrdagdelen@zrdagdelen-MacBookPro: ~
FIND(1) General Commands Manual FIND(1)
NAME
    find - search for files in a directory hierarchy
SYNOPSIS
    find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [ex-
    pression]
DESCRIPTION
    This manual page documents the GNU version of find. GNU find searches
    the directory tree rooted at each given starting-point by evaluating
    the given expression from left to right, according to the rules of
    precedence (see section OPERATORS), until the outcome is known (the
    left hand side is false for and operations, true for or), at which
```

Рис. 4.18: Чтение документации

Вывела имена всех директорий, имеющиххся 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. fig. 4.19). Утилита -a позволит увидеть размер всех файлов, а не только диреткорий.

```
zrdagdelen@zrdagdelen-MacBookPro: ~ % find -type d
.
./pandoc-3.1.12.2
./pandoc-3.1.12.2/share
./pandoc-3.1.12.2/share/man
./pandoc-3.1.12.2/share/man/man1
./pandoc-3.1.12.2/bin
./.cache
./.cache/mesa_shader_cache
./.cache/mesa_shader_cache/0d
./.cache/mesa_shader_cache/ac
./.cache/mesa_shader_cache/b9
./.cache/mesa_shader_cache/39
./.cache/mesa_shader_cache/68
./.cache/mesa_shader_cache/c4
./.cache/mesa_shader_cache/96
./.cache/mesa_shader_cache/ca
```

Рис. 4.19: Название рисунка

5 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.

6 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 `»`. Этот знак `>` - перенаправление ввода/вывода, а `»` - перенаправление 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? 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. 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 - 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

9. Можно ли по контексту (содержанию) найти файл? Если да, то как? `find / -type f -exec grep -H 'текстДляПоиска' {} ;`
10. Как определить объем свободной памяти на жёстком диске? С помощью команды `df -h`.
11. Как определить объем вашего домашнего каталога? С помощью команды `du -s`.
12. Как удалить зависший процесс? С помощью команды `kill%` номер задачи.