

Отчет по выполнению лабораторной работы №9

Операционные системы

Федорова Анжелика Игоревна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
4	Выполнение лабораторной работы	9
5	Выводы	14
6	Ответы на контрольные вопросы	15
7	Выводы	18
8	Список литературы	19

Список иллюстраций

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

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 директо-

рий, имею- щихся 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:001).

```
aifedorova@aifedorova:~$ sudo ls -lR /etc > file.txt
```

Рис. 4.1: Запись в файл

Проверила, что в файл записались нужные значения с помощью команды cat (рис. fig. 4.2).

```
aifedorova@aifedorova:~$ cat file.txt
/etc:
итого 1464
drwxr-xr-x. 1 root root      126 ноя  1 04:08 abrt
-rw-r--r--. 1 root root       16 фев 13 20:54 adjtime
-rw-r--r--. 1 root root    1529 июл 25  2023 aliases
drwxr-xr-x. 1 root root       70 янв 29 03:00 alsa
drwxr-xr-x. 1 root root    1536 фев 25 22:20 alternatives
drwxr-xr-x. 1 root root       56 ноя  1 04:08 appendo
```

Рис. 4.2: Вывод содержимого файла

Добавила в созданный файл имена файлов из домашнего каталога, используя перенаправление “>” в режиме добавления (рис. fig:003).

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

Рис. 4.3: Добавление данных в файл

Вывела на экран имена всех файлов, имеющих расширение “.conf” с помощью утилиты grep (рис. fig:004).

```
aifedorova@aifedorova:~$ grep .conf file.txt
-rw-r--r--. 1 root root      269 июл 19  2023 anthy-unicode.conf
-rw-r--r--. 1 root root      833 фев 10  2023 appstream.conf
-rw-r--r--. 1 root root        55 янв 29 03:00 asound.conf
-rw-r--r--. 1 root root    30583 дек 20 03:00 brltty.conf
drwxr-xr-x. 1 root root         0 янв 17 03:00 chkconfig.d
-rw-r--r--. 1 root root     1372 дек  5 03:00 chrony.conf
drwxr-xr-x. 1 root root        18 ноя  1 04:06 dconf
-rw-r--r--. 1 root root     1174 сен 20  2023 dleyna-server-service.conf
-rw-r--r--. 1 root root    28601 янв 12 03:00 dnsmasq.conf
-rw-r--r--. 1 root root       117 ноя 16 03:00 dracut.conf
drwxr-xr-x. 1 root root         0 ноя 16 03:00 dracut.conf.d
```

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:005).

```
aifedorova@aifedorova:~$ grep .conf file.txt > conf.txt
aifedorova@aifedorova:~$ head conf.txt
-rw-r--r--. 1 root root      269 июл 19  2023 anthy-unicode.conf
-rw-r--r--. 1 root root      833 фев 10  2023 appstream.conf
-rw-r--r--. 1 root root        55 янв 29 03:00 asound.conf
-rw-r--r--. 1 root root    30583 дек 20 03:00 brltty.conf
drwxr-xr-x. 1 root root         0 янв 17 03:00 chkconfig.d
-rw-r--r--. 1 root root     1372 дек  5 03:00 chrony.conf
```

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:006). Но таким образом я получаю информацию даже о файлах из подкаталогов домашнего каталога.

```
aifedorova@aifedorova:~$ find ~ -name "с*" -print
/home/aifedorova/.mozilla/firefox/gjaqarar.default-release/crashes
/home/aifedorova/.mozilla/firefox/gjaqarar.default-release/compatibility.ini
/home/aifedorova/.mozilla/firefox/gjaqarar.default-release/cookies.sqlite
/home/aifedorova/.mozilla/firefox/gjaqarar.default-release/cert9.db
```

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:007).

```
aifedorova@aifedorova:~$ ls -lR | grep c*
-rw-r--r--. 1 aifedorova aifedorova 54375 map 27 22:00 conf.txt
aifedorova@aifedorova:~$
```

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).

```
aifedorova@aifedorova:~$ find /etc -name "h*" -print
```

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:009).

```
aifedorova@aifedorova:~$ find ~ -name "log*" > ~/logfile &
[1] 110680
```

Рис. 4.9: Создание фонового процесса

Удаляю созданный файл (рис. fig. 4.10).

```
[1]+  Завершён      find ~ -name "log*" > ~/logfile
aifedorova@aifedorova:~$ rm logfile
```

Рис. 4.10: Удаление файла

Запускаю в консоли в фоновом режиме (с помощью символа `&`) редактор `mouesrad`, потому что редактора `gedit` у меня, к сожалению, но работают они идентично (рис. fig. 4.11).

```
aifedorova@aifedorova:~$ gedit &
```

Рис. 4.11: Создание фонового процесса

С помощью утилиты ps определяю идентификатор процесса mousepad, его значение 3913 (рис. fig. 4.12). Также мы можем определить идентификатор с помощью pgrep.

```
aifedorova@aifedorova:~$ ps aux | grep gedit
aifedor+ 110759  0.0  1.9 930452 74888 pts/0    Sl   23:06   0:01 gedit
aifedor+ 111120  0.0  0.0 222456  2176 pts/0    S+   23:33   0:00 grep --color=auto gedit
```

Рис. 4.12: Поиск идентификатора процесса

Прочитала справку команды kill и использую команду kill и идентификатор процесса, чтобы его удалить (рис. fig. 4.13). Заметила, как у меня закрылась программа gedit.

```
aifedorova@aifedorova:~$ man kill
aifedorova@aifedorova:~$ kill 110759
```

Рис. 4.13: Удаление процесса

Прочитала документацию про функции df и du. Теперь я использую утилиту df с файлом conf.txt. (fig:014). Эта утилита нам нужна, чтобы выяснить, сколько свободного места есть у нашей системы.

```
aifedorova@aifedorova:~$ df conf.txt
Файловая система 1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
/dev/sda3          32587776      19249248  12066064          62% /home
```

Рис. 4.14: Утилита df

Использую утилиту du. Она нужна чтобы просмотреть, сколько места занимают файлы в определенной директории и найти самые большие из них (рис. fig:017).

```
aifedorova@aifedorova:~$ du conf.txt
56      conf.txt
```

Рис. 4.15: Утилита du

Прочитала документацию о команде find (рис. fig:018).

Вывела имена всех директорий, имеющихя 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:016). Утилита `find` позволит увидеть размер всех файлов, а не только директорий.

```
/home/aifedorova/play/games  
/home/aifedorova/Pictures  
/home/aifedorova/Pictures/Снимки экрана  
aifedorova@aifedorova:~$ find ~ -type d
```

Рис. 4.16: Вывод директорий

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.

7 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.

8 Список литературы

Лабораторная работа №9