

Лабораторная работа № 8

Простейший вариант

Лупупа Чилеше

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Контрольные вопросы	15
3	Выводы	21
	Список литературы	22

Список иллюстраций

2.1	file.txt	7
2.2	conf.txt	8
2.3	начинавшиеся с символа с	9
2.4	начинавшиеся с символа с	10
2.5	начинающиеся с символа h	11
2.6	~/logfile	11
2.7	gedit	12
2.8	df	13
2.9	du	14
2.10	find	15
2.11	find	15

Список таблиц

1 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.

2 Выполнение лабораторной работы

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.

```
[lchileshe@lchileshe ~]$ las /etc > file.txt
bash: las: command not found
[lchileshe@lchileshe ~]$ ls /etc > file.txt
[lchileshe@lchileshe ~]$ cat file.txt
abrt
adjtime
aliases
alsa
alternatives
anaconda
anacrontab
appstream.conf
asound.conf
at.deny
audit
authselect
avahi
bash_completion.d
bashrc
bindresvport.blacklist
binfmt.d
bluetooth
chromium
chrony.conf
cifs-utils
credstore
credstore.encrypted
cron.d
cron.daily
cron.deny
cron.hourly
cron.monthly
crontab
cron.weekly
crypto-policies
```

Рис. 2.1: file.txt

3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего

```
[lchileshe@lchileshe ~]$ grep .conf file.txt
appstream.conf
asound.conf
chrony.conf
dconf
dnsmasq.conf
dracut.conf
dracut.conf.d
fprintd.conf
fuse.conf
host.conf
idmapd.conf
imv_config
ipsec.conf
kdump.conf
krb5.conf
krb5.conf.d
ld.so.conf
ld.so.conf.d
libaudit.conf
libuser.conf
locale.conf
logrotate.conf
```

4. запишите их в новый текстовый файл conf.txt. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа c? Предложите несколько вариантов, как это сделать.

```
[lchileshe@lchileshe ~]$ grep "\.conf " file.txt > conf.txt
[lchileshe@lchileshe ~]$
```

Рис. 2.2: conf.txt


```
[lchileshe@lchileshe ~]$ ls -R | grep ^c
conf.txt
chezmoi
config
content
chris-montgomery-smgTvepind4-unsplash.jpg
ci - cd
conference-paper
cite.bib
conference-paper.pdf
cite.bib
categories
category
ci---cd
css
chroma
```

Рис. 2.3: начинавшиеся с символа с

```
/home/lchileshe/work/Blog/public/publication/journal-article/ci
te.bib
/home/lchileshe/work/Blog/public/ci---cd
/home/lchileshe/work/LupupaChileshe.github.io/.git/hooks/commit
-msg.sample
/home/lchileshe/work/LupupaChileshe.github.io/.git/config
/home/lchileshe/git-extended/.git/hooks/commit-msg.sample
/home/lchileshe/git-extended/.git/objects/a1/ca712dd6bae3638426
c0564d0db5b8aa6b6b36
/home/lchileshe/git-extended/.git/config
/home/lchileshe/.password-store/.git/hooks/commit-msg.sample
/home/lchileshe/.password-store/.git/config
/home/lchileshe/.password-store/bin/chezmoi
/home/lchileshe/bin/chezmoi
/home/lchileshe/conf.txt
[lchileshe@lchileshe ~]$
```

Рис. 2.4: начинавшиеся с символа с

5. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.

```

/home/lchileshe/.local/share/pnpm/global/5/.pnpm/conventional-c
hangelog-angular@7.0.0/node_modules/conventional-changelog-angu
lar/templates/header.hbs
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/conventional-c
hangelog-core@7.0.0/node_modules/hosted-git-info
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/conventional-c
hangelog-core@7.0.0/node_modules/conventional-changelog-core/ho
sts
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/hosted-git-inf
o@7.0.1
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/hosted-git-inf
o@7.0.1/node_modules/hosted-git-info
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/hosted-git-inf
o@7.0.1/node_modules/hosted-git-info/lib/hosts.js
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/normalize-pack
age-data@6.0.0/node_modules/hosted-git-info
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/is-core-module
@2.13.1/node_modules/hasown
/home/lchileshe/.local/share/pnpm/global/5/.pnpm/hasown@2.0.1

```

Рис. 2.5: начинающиеся с символа h

6. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.

```

[lchileshe@lchileshe ~]$ find ~ -name "log*" -print >> logfile
&
[1] 2795
[lchileshe@lchileshe ~]$

```

Рис. 2.6: ~/logfile

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?

```
[lchileshe@lchileshe ~]$ ps | grep gedit
[1]+  Done                  gedit
[lchileshe@lchileshe ~]$
```

Рис. 2.7: gedit

10. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.

```
[lchileshe@lchileshe ~]$ kill 3053
[lchileshe@lchileshe ~]$
```

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

SYNOPSIS

```
df [OPTION]... [FILE]...
```

DESCRIPTION

This manual page documents the GNU version of `df`. `df` displays the amount of space available on the file system containing each file name argument. If no file name is given, the space available on all currently mounted file systems is shown. Space is shown in 1K blocks by default, unless the environment variable `POSIXLY_CORRECT` is set, in which case 512-byte blocks are used.

If an argument is the absolute file name of a device node containing a mounted file system, `df` shows the space available on that file system rather than on the file system containing the device node. This version of `df` cannot show the space available on unmounted file systems, because on most kinds of systems doing so requires very nonportable intimate knowledge of file system structures.

Рис. 2.8: `df`

```
NAME
    du - estimate file space usage

SYNOPSIS
    du [OPTION]... [FILE]...
    du [OPTION]... --files0-from=F

DESCRIPTION
    Summarize device usage of the set of FILEs, recursively for directories.

    Mandatory arguments to long options are mandatory for short options too.

    -0, --null
        end each output line with NUL, not newline

    -a, --all
        write counts for all files, not just directories

    --apparent-size
        print apparent sizes rather than device usage; although the apparent size is usually smaller, it may be larger due to holes in ('sparse') files, internal fragmentation, indirect blocks, and the like

    -B, --block-size=SIZE
        scale sizes by SIZE before printing them; e.g., '-BM' prints sizes in units of 1,048,576 bytes; see SIZE format below
```

Рис. 2.9: du

12. Воспользовавшись справкой команды find, выведите имена всех директо-

рий, имеющихся в вашем домашнем каталоге.

```
[lchileshe@lchileshe ~]$ find ~ -name "*" -type d -print
```

Рис. 2.10: find

```
/home/lchileshe/.password-store/.git/refs/remotes
/home/lchileshe/.password-store/.git/refs/remotes/origin
/home/lchileshe/.password-store/.git/objects
/home/lchileshe/.password-store/.git/objects/pack
/home/lchileshe/.password-store/.git/objects/info
/home/lchileshe/.password-store/.git/objects/8f
/home/lchileshe/.password-store/.git/objects/5b
/home/lchileshe/.password-store/.git/objects/d0
/home/lchileshe/.password-store/.git/objects/f9
/home/lchileshe/.password-store/.git/objects/a0
/home/lchileshe/.password-store/.git/objects/11
/home/lchileshe/.password-store/.git/objects/19
/home/lchileshe/.password-store/.git/objects/6b
/home/lchileshe/.password-store/.git/objects/74
/home/lchileshe/.password-store/.git/objects/03
/home/lchileshe/.password-store/.git/objects/99
/home/lchileshe/.password-store/.git/logs
/home/lchileshe/.password-store/.git/logs/refs
/home/lchileshe/.password-store/.git/logs/refs/heads
/home/lchileshe/.password-store/.git/logs/refs/remotes
/home/lchileshe/.password-store/.git/logs/refs/remotes/origin
/home/lchileshe/.password-store/bin
/home/lchileshe/.bashrc.d
/home/lchileshe/bin
/home/lchileshe/reports
/home/lchileshe/reports/monthly
/home/lchileshe/reports/monthly/monthly
/home/lchileshe/monthly
/home/lchileshe/ski.places
/home/lchileshe/ski.places/equipment
/home/lchileshe/ski.places/plans
/home/lchileshe/australia
/home/lchileshe/play
/home/lchileshe/play/games
```

Рис. 2.11: find

2.1 Контрольные вопросы

1. Какие потоки ввода вывода вы знаете?
2. Объясните разницу между операцией > и ».

- Оператор > перезаписывает существующий файл или создает новый файл, если файл с указанным именем отсутствует. Это означает, что если файл уже существует, его содержимое будет полностью заменено выводом команды
- Оператор » добавляет вывод команды в конец существующего файла или создает новый файл, если файл с указанным именем отсутствует. Это позволяет сохранять предыдущие данные в файле и добавлять к ним новые данные.

3. Что такое конвейер?

- Конвейер (pipeline) в терминологии операционных систем семейства Unix — это механизм, позволяющий связывать несколько процессов в последовательность, где вывод предыдущего процесса становится входными данными для следующего процесса.

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

- Процесс и программа — это два ключевых понятия в компьютерных науках, которые часто путают, но имеют различные значения.
- Программа — это статический набор инструкций, написанных на языке программирования, который может быть выполнен компьютером. Программа содержит код, который определяет, что должно быть сделано, но не выполняется сама по себе. Программа может быть загружена в оперативную память и выполнена процессором, но для этого необходимо создать процесс.
- Процесс — это активное выполнение программы. Когда программа запускается, операционная система создает процесс, который включает в себя код программы, а также ресурсы, такие как память, файловые дескрипторы и идентификаторы процессов. Процесс является исполняемым объек-

том, который может быть запланирован и выполнен процессором. Процесс может иметь один или несколько потоков выполнения, которые могут работать параллельно в рамках одного процесса.

5. Что такое PID и GID?

- PID (Process ID) — это уникальный идентификатор, присваиваемый каждому процессу в системе. PID используется для идентификации процесса в системе и для управления им, например, для завершения процесса. Каждый процесс в системе имеет свой уникальный PID, который позволяет операционной системе и пользователям отслеживать и управлять процессами
- GID (Group ID) — это идентификатор группы, который используется для определения, к какой группе пользователей принадлежит процесс или пользователь. GID используется для управления доступом к ресурсам и файлам, а также для определения, какие права доступа имеет процесс или пользователь к этим ресурсам. В Unix-подобных системах, GID 0 зарезервирован для суперпользователя (root), а диапазон GID 1–99 зарезервирован для системных и приложений. GID 100 и выше обычно выделяются для групп пользователей

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

- Задачи в контексте операционных систем, таких как Linux, обычно относятся к процессам или задачам, которые выполняются в системе. Они могут быть активными (запущены и выполняются) или ожидающими (приостановлены и не выполняются). Управление задачами включает в себя запуск, остановку, приостановку, возобновление и мониторинг процессов в системе.

Для управления задачами в командной строке Linux используются различные команды, включая:

ps: Показывает список запущенных процессов на компьютере.

kill: Отправляет сигнал одному или нескольким процессам, что может привести к их остано-

jobs: Позволяет просмотреть список собственных задач сеанса оболочки.

bg: Выполняет задачу в фоновом режиме, позволяя продолжить работу в командной строке

fg: Возвращает задачу из фонового режима, делая ее активной в текущем терминале.

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

- Утилиты top и htop являются популярными инструментами командной строки для мониторинга системы в Linux. Они предоставляют динамический, реальный вид работающей системы, позволяя пользователям видеть и управлять процессами в реальном времени.

top:

top — это старый и базовый инструмент, который поставляется с большинством дистрибутивов Linux. Он предоставляет информацию о системе и список процессов или потоков, основанных на использовании CPU, которые в данный момент управляются ядром Linux.

htop:

htop — это более новый и улучшенный вариант top, который включает в себя множество улучшений. Он поддерживает взаимодействие с мышью, использует цвета в своем выводе и предоставляет визуальные индикации об использовании процессора, памяти и подкачки.

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

- Команда find в Linux — это мощный инструмент для поиска файлов и директорий в файловой системе. Она позволяет выполнять поиск по различным критериям, таким как имя файла, тип файла, время изменения, размер и другие атрибуты.

find [directory to search] [options] [expression]

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

- Да, можно найти файл по его содержанию, используя команду `find` в сочетании с командой `grep`. Это позволяет выполнить поиск по всей файловой системе или в определенной директории для файлов, содержащих указанный текст.

10. Как определить объем свободной памяти на жёстком диске?

- Чтобы определить объем свободной памяти на жестком диске в Linux, можно использовать следующие команды:

Команда `df`: `df` означает “файловая система диска” и позволяет отобразить свободное место на диске. Примеры использования: `df` - покажет информацию в стандартном формате. `df -h` - отобразит информацию в удобном для человека формате (например, в гигабайтах). `df -m` - отобразит информацию в мегабайтах. `df -k` - отобразит информацию в килобайтах. `df -T` - добавит колонку с типом файловой системы. `df -ht /home` - покажет информацию о конкретной файловой системе (в данном случае, о файловой системе, монтированной в `/home`) в удобном формате.

Команда `du`: `du` используется для анализа использования дискового пространства файлами и папками. Примеры использования: `du /home/user/Desktop/` - покажет информацию об использовании дискового пространства файлами и папками на рабочем столе пользователя. `du -h /home/user/Desktop/` - отобразит информацию в более удобном для человека формате. `du -sh /home/user/Desktop/` - покажет полный объем папки `Desktop`. `du -m /home/user/Desktop/` - отобразит информацию в мегабайтах. `du -h -time /home/user/Desktop/` - даст информацию, основанную на последнем изменении файлов и папок.

11. Как определить объем вашего домашнего каталога?

- Чтобы определить объем вашего домашнего каталога в Linux, вы можете использовать команду `du`. Эта команда позволяет анализировать использование дискового пространства файлами и папками.

12. Как удалить зависший процесс?

- Чтобы удалить зависший процесс в Linux, вы можете использовать команду `kill` с идентификатором процесса (PID).

3 Выводы

Здесь кратко описываются итоги проделанной работы.

Список литературы