

Отчёт по лабораторной работе №8

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Джаллох Ишмаил

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	14
5	Ответы на контрольные вопросы	15

Список иллюстраций

3.1	Запись в файл	7
3.2	Первые 8 файлов в file.txt	7
3.3	Добавление файлов из домашнего каталога	7
3.4	файлы с расширением .conf	8
3.5	добавление файлов с расширением .conf	8
3.6	файлы в домашнем каталоге начинающихся с “с”	9
3.7	поиск файла используя grep	9
3.8	файлы в etc начинающихся с “h”	10
3.9	Создание фонового режима	10
3.10	удаление logfile	11
3.11	запуск gedit в фоновом режиме	11
3.12	идентификатор процесса gedit	11
3.13	Другой способ нахождения идентификатора процесса	11
3.14	завершения процесса gedit	11
3.15	справка команды df	12
3.16	справка команды du	12
3.17	df -vi	12
3.18	du -a	13
3.19	Поиск директорий	13
3.20	результаты find ~ -type d	13

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 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 Выполнение лабораторной работы

Вошел в систему под моим именем, открыл терминал и записал в файле file.txt названия файлов, содержащихся в каталоге /etc с помощью `ls -lR /etc > file.txt` :

```
ijalloh@ijalloh:~$ sudo ls -lR /etc > file.txt
[sudo] password for ijalloh:
ijalloh@ijalloh:~$
```

Рис. 3.1: Запись в файл

С помощью head я проверяю, что в файл записался названия файлов, содержащихся в каталоге /etc:

```
ijalloh@ijalloh:~$ head file.txt
/etc:
total 1460
drwxr-xr-x. 1 root          root          126 Nov 20 03:00 abr
t
-rw-r--r--. 1 root          root           16 Feb 28 04:19 adj
time
-rw-r--r--. 1 root          root        1529 Dec 27 03:00 ali
ases
drwxr-xr-x. 1 root          root           70 Nov 14 03:00 als
a
drwxr-xr-x. 1 root          root        1536 Mar  8 12:38 alt
ernatives
drwxr-xr-x. 1 root          root           56 Oct 24 17:54 ana
conda
-rw-r--r--. 1 root          root          269 Aug 16 2024 ant
hy-unicode.conf
-rw-r--r--. 1 root          root           55 Nov 15 03:00 aso
und.conf
ijalloh@ijalloh:~$
```

Рис. 3.2: Первые 8 файлов в file.txt

В file.txt добавляю названия файлов, из домашнего каталога используя `ls -lR /etc » file.txt`:

```
ijalloh@ijalloh:~$ sudo ls -lR ~/ >> file.txt
[sudo] password for ijalloh:
```

Рис. 3.3: Добавление файлов из домашнего каталога

Вывожу имена всех файлов из file.txt, имеющих расширение .conf с помощью grep:

```
ijalloheijalloh:~$ grep .config file.txt
drwxr-xr-x. 1 root      root      0 Dec 18 03:00 chk
config.d
-rw-r--r--. 1 root      root      767 Jan  2 03:00 net
config
drwxr-xr-x. 1 root      root      26 Oct 24 17:50 pkg
config
drwxr-xr-x. 1 root      root      384 Mar 26 00:54 sys
config
/etc/chkconfig.d:
-rw-r--r--. 1 root root  8 Oct 24 17:49 config
lrwxrwxrwx. 1 root root 43 Feb 28 05:28 bind.config -> /usr/share/crypto-policies/DEFAULT/bind.txt
lrwxrwxrwx. 1 root root 45 Feb 28 05:28 gnutls.config -> /usr/share/crypto-policies/DEFAULT/gnutls.txt
```

Рис. 3.4: файлы с расширением .conf

Затем запишу их в новый текстовый файл conf.txt (grep .conf file.txt > conf.txt) и проверю с помощью head:

```
ijalloheijalloh:~$ grep .config file.txt > conf.txt
ijalloheijalloh:~$ head conf.txt
drwxr-xr-x. 1 root      root      0 Dec 18 03:00 chk
config.d
-rw-r--r--. 1 root      root      767 Jan  2 03:00 net
config
drwxr-xr-x. 1 root      root      26 Oct 24 17:50 pkg
config
drwxr-xr-x. 1 root      root      384 Mar 26 00:54 sys
config
/etc/chkconfig.d:
-rw-r--r--. 1 root root  8 Oct 24 17:49 config
lrwxrwxrwx. 1 root root 43 Feb 28 05:28 bind.config -> /usr/share/crypto-policies/DEFAULT/bind.txt
lrwxrwxrwx. 1 root root 45 Feb 28 05:28 gnutls.config -> /usr/share/crypto-policies/DEFAULT/gnutls.txt
```

Рис. 3.5: добавление файлов с расширением .conf

Чтобы определить, какие файлы в домашнем каталоге имеют имена, начинавшиеся с символа “с”, использую find ~ -name “с” print ; ~ обозначается домашний каталог, -name (имя файлов) “с” строка символов, определяющая имя файла и print выводит результаты на экране:


```

/home/ijalloh/work/blog/.git/hooks/commit-msg.sample
/home/ijalloh/work/blog/.git/config
/home/ijalloh/work/blog/config
/home/ijalloh/work/blog/content
/home/ijalloh/work/blog/content/publication/conference-paper
/home/ijalloh/work/blog/content/publication/conference-paper/cite.bib
/home/ijalloh/work/blog/content/publication/conference-paper/conference-paper.pdf
/home/ijalloh/work/blog/content/publication/journal-article/cite.bib
/home/ijalloh/.stack/config.yaml
/home/ijalloh/.npm/_cacache/content-v2
/home/ijalloh/.npm/_cacache/content-v2/sha512/bc/a6/cce3b37bb4e021eac03a79ea866
89614e1b400b210197f8c8bab512a8507b7c05ab0bf6bf251238a5ef6ee2d4e2a6de262a37a9620
81c678123cddc22378
/home/ijalloh/.npm/_cacache/content-v2/sha512/ce
/home/ijalloh/.npm/_cacache/content-v2/sha512/c6
/home/ijalloh/.npm/_cacache/content-v2/sha512/0e/cc
/home/ijalloh/.npm/_cacache/content-v2/sha512/0e/cc/c780c74ed62f1f1ba74fbf9d43e
7a2a2f5095f1874852f6625c06f5277c0644a5f09d7007027350bf4582d273b015a041058423c0d

```

Рис. 3.6: файлы в домашнем каталоге начинающихся с “с”

Также можно это действие выполнить используя `ls -lR | grep “с*”`

```

ijalloh@ijalloh:~$ ls -lR | grep c*
-rw-r--r--. 1 ijalloh ijalloh 22590 Apr  3 12:50 conf.txt

```

Рис. 3.7: поиск файла используя grep

с помощью `find /etc -name “h*” -print`, вывожу файлы из каталога /etc, начинающиеся с символа h:

```

ijalloheijalloh:~$ sudo find /etc -name "h*" -print
[sudo] password for ijalloh:
/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
/etc/firewalld/helpers
/etc/hp
/etc/hp/hplip.conf
/etc/httpd
/etc/httpd/conf/httpd.conf
/etc/libibverbs.d/hfi1verbs.driver
/etc/libibverbs.d/hns.driver
/etc/logrotate.d/httpd
/etc/nvme/hostnqn
/etc/nvme/hostid
/etc/sane.d/dll.d/hpaio
/etc/sane.d/hp.conf
/etc/sane.d/hp3900.conf
/etc/sane.d/hp4200.conf
/etc/sane.d/hp5400.conf
/etc/sane.d/hpsj5s.conf
/etc/sane.d/hs2p.conf
/etc/sysconfig/htcacheclean
/etc/systemd/system/httpd.service.d
/etc/udev/hwdb.d
/etc/udev/hwdb.bin
/etc/host.conf
/etc/hosts
/etc/hostname
/etc/mercurial/hgrc.d

```

Рис. 3.8: файлы в etc начинающихся с “h”

В фоновом режиме запускаю процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log:

```

ijalloheijalloh:~$ sudo find ~ -name "log*" -print > logfile &
[1] 7464
ijalloheijalloh:~$
[1]+  Done                  sudo find ~ -name "log*" -print > logfile
ijalloheijalloh:~$ █

```

Рис. 3.9: Создание фонового режима

Удаляю созданный logfile и проверяю:

```

ijalloheijalloh:~$ ls
abc1      Desktop  file.txt  LICENSE  Music    Public   texput.log
australia Documents fun      logfile  my_os    reports  Videos
bin        Downloads git-extended may      Pictures ski.places work
conf.txt  feathers go        monthly  play     Templates
ijalloheijalloh:~$ rm logfile
ijalloheijalloh:~$ ls
abc1      Desktop  file.txt  LICENSE  my_os    reports  Videos
australia Documents fun      may      Pictures ski.places work
bin        Downloads git-extended monthly  play     Templates
conf.txt  feathers go        Music    Public   texput.log
ijalloheijalloh:~$

```

Рис. 3.10: удаление logfile

Запускаю из консоли в фоновом режиме редактор gedit указывая &:

```

ijalloh@ijalloh:~$ gedit &
[1] 7778
ijalloh@ijalloh:~$

```

Рис. 3.11: запуск gedit в фоновом режиме

Используя команду ps, конвейер и фильтр grep, определяю идентификатор процесса gedit (3576):

```

ijalloheijalloh:~$ ps aux | grep gedit
ijalloh   7778  2.1  1.6 780344 65964 pts/0    Sl   13:08   0:01 gedit
ijalloh   7838  0.0  0.0 230340  2236 pts/0    S+   13:08   0:00 grep --color
=auto gedit

```

Рис. 3.12: идентификатор процесса gedit

```

ijalloheijalloh:~$ ps aux | grep gedit | grep -v grep
ijalloh   7778  1.0  1.6 780344 66092 pts/0    Sl   13:08   0:01 gedit
ijalloheijalloh:~$

```

Рис. 3.13: Другой способ нахождения идентификатора процесса

С помощью man прочитала справку команды kill и использую её для завершения процесса gedit:

```

ijalloheijalloh:~$ man kill
ijalloheijalloh:~$ kill 7778
ijalloheijalloh:~$

```

Рис. 3.14: завершения процесса gedit

С помощью man прочитала справку команд df и du:

```

NAME
    df - report file system space usage

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 cur-
    rently 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.

```

Рис. 3.15: справка команды 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 directo-
    ries.

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

```

Рис. 3.16: справка команды du

Используя `df -vi` я вывожу информацию об инодах и вижу сколько свободного места у моей системы:

```

ijalloh@ijalloh:~$ df -vi
Filesystem      Inodes  IUsed   IFree IUse% Mounted on
/dev/sda3         0         0      0     - /
devtmpfs        495371    564  494807    1% /dev
tmpfs           499473      6  499467    1% /dev/shm
tmpfs           819200   1097  818103    1% /run
tmpfs           1024      2   1022    1% /run/credentials/systemd-journald.se
rvice
tmpfs           1024      2   1022    1% /run/credentials/systemd-network-gen
erator.service
tmpfs           1024      2   1022    1% /run/credentials/systemd-udev-load-c
redentials.service
tmpfs           1024      2   1022    1% /run/credentials/systemd-sysctl.serv
ice
tmpfs           1024      2   1022    1% /run/credentials/systemd-tmpfiles-se
tmp-dev-early.service
tmpfs           1024      2   1022    1% /run/credentials/systemd-tmpfiles-se
tmp-dev.service
tmpfs          1048576    52 1048524    1% /tmp
/dev/sda3         0         0      0     - /home
/dev/sda2        65536    405  65131    1% /boot
tmpfs           1024      2   1022    1% /run/credentials/systemd-tmpfiles-se

```

Рис. 3.17: df -vi

Используя `du -a` я вижу сколько места занимают файлы в директории Загрузки:

```

ijalloheijalloh:~$ du -a Downloads
31244 Downloads/pandoc-3.1.12.2-linux-amd64.tar.gz
4 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/bin/pandoc-server
4 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/bin/pandoc-lua
148880 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/bin/pandoc
148888 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/bin
80 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share/man/man1/pa
ndoc.1.gz
4 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share/man/man1/pa
ndoc-lua.1.gz
8 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share/man/man1/pa
ndoc-server.1.gz
92 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share/man/man1
92 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share/man
92 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2/share
148980 Downloads/pandoc-3.1.12.2-linux-amd64/pandoc-3.1.12.2
148980 Downloads/pandoc-3.1.12.2-linux-amd64
8544 Downloads/pandoc-crossref-Linux.tar.xz
52764 Downloads/hugo
16 Downloads/README.md
12 Downloads/LICENSE
18236 Downloads/hugo_extended_0.145.0_Linux-64bit.tar.gz
18236 Downloads/hugo_extended_0.145.0_Linux-64bit(1).tar.gz
18236 Downloads/hugo_extended_0.145.0_Linux-64bit(2).tar.gz
52812 Downloads/hugo_extended_0.145.0_Linux-64bit(2)/hugo

```

Рис. 3.18: du -a

Воспользовавшись справкой команды find и аргумент d, вывожу всех директо-
рий, имеющих в домашнем каталоге:

```

ijalloh@ijalloh:~$ find ~ -type d
/home/ijalloh

```

Рис. 3.19: Поиск директорий

```

/home/ijalloh/.password-store
/home/ijalloh/.password-store/.git
/home/ijalloh/.password-store/.git/hooks
/home/ijalloh/.password-store/.git/info
/home/ijalloh/.password-store/.git/refs
/home/ijalloh/.password-store/.git/refs/heads
/home/ijalloh/.password-store/.git/refs/tags
/home/ijalloh/.password-store/.git/objects
/home/ijalloh/.password-store/.git/objects/pack
/home/ijalloh/.password-store/.git/objects/info
/home/ijalloh/.password-store/.git/objects/a5
/home/ijalloh/.password-store/.git/objects/83
/home/ijalloh/.password-store/.git/objects/8b
/home/ijalloh/.password-store/.git/objects/f9
/home/ijalloh/.password-store/.git/objects/6a
/home/ijalloh/.password-store/.git/objects/c0
/home/ijalloh/.password-store/.git/objects/89
/home/ijalloh/.password-store/.git/objects/c4
/home/ijalloh/.password-store/.git/objects/59
/home/ijalloh/.password-store/.git/objects/23
/home/ijalloh/.password-store/.git/objects/05
/home/ijalloh/.password-store/.git/objects/be
/home/ijalloh/.password-store/.git/objects/c3
/home/ijalloh/.password-store/.git/objects/54
/home/ijalloh/.password-store/.git/logs

```

Рис. 3.20: результаты find ~ -type d

4 Выводы

При выполнении данной работы я ознакомился с инструментами поиска файлов и фильтрации текстовых данных. Также приобрел практические навыки по управлению процессами и по проверке использования диска по обслуживанию файловых систем.

5 Ответы на контрольные вопросы

1. `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.
 - Перенаправление вывода (`stdout`) в файл “filename”, » файл открывается в режиме добавления.
3. Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей.
4. Программа - это набор инструкций, который позволяет ЦПУ выполнять определенную задачу, в то время как процесс - это исполняемая программа.
5. 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. Запущенные фоном программы называются задачами (`jobs`). Ими можно управлять с помощью команды `jobs`, которая выводит список запущенных в данный момент задач.
7. 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.

8. Команда `find` - это команда для поиска файлов и каталогов на основе специальных условий. Ее можно использовать в различных обстоятельствах, например, для поиска файлов по разрешениям, группам, типу, размеру и другим подобным критериям. Утилита `find` предустановлена по умолчанию во всех Linux дистрибутивах. Команда `find` имеет такой синтаксис: `find [папка] [параметры] критерий шаблон [действие]` Пример: `find /etc -name "p*" -print`
9. `find / -type f -exec grep -H 'текстДляПоиска' {} ;`
10. `df -h`.
11. `du -s`.
12. `kill%` номер задачи.