

Отчёт по лабораторной работе №8

**Поиск файлов. Перенаправление ввода-вывода. Просмотр
запущенных процессов**

Бельчуг Александр Константинович

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Вывод	13
4	Контрольные вопросы	14

Список иллюстраций

2.1	Запись в файл	7
2.2	Поиск расширения .conf	8
2.3	Поиск файлов	8
2.4	Поиск файлов	9
2.5	Фоновый запуск процесса	9
2.6	Фоновый запуск и завершение процесса	10
2.7	Справка по команде df	10
2.8	Запуск команды df	10
2.9	Справка по команде du	11
2.10	Запуск команды du	11
2.11	Поиск директорий	12

Список таблиц

1 Цель работы

Ознакомление с инструментами поиска файлов и фильтрации текстовых данных. Приобретение практических навыков: по управлению процессами, по проверке использования диска и обслуживанию файловых систем.

2 Выполнение лабораторной работы

1 Включаем компьютер, и заходим в учетную запись.

2 Запишем в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишем в этот же файл названия файлов, содержащихся в нашем домашнем каталоге.

```
[akbeljchug@vbox ~]$ ls /etc/ > file.txt
[akbeljchug@vbox ~]$ ls >> file.txt
[akbeljchug@vbox ~]$ cat file.txt
abrt
adjtime
aliases
alsa
alternatives
anaconda
anacrontab
asound.conf
at.deny
audit
authselect
avahi
bash_completion.d
bashrc
bindresvport.blacklist
binfmt.d
bluetooth
chrony.conf
cifs-utils
credstore
credstore.encrypted
cron.d
cron.daily
cron.deny
cron.hourly
cron.monthly
crontab
cron.weekly
crypto-policies
crypttab
csh.cshrc
csh.login
cups
cupshelpers
dbus-1
dconf
debuginfod
default
depmod.d
dhcp
DIR_COLORS
```

Рис. 2.1: Запись в файл

3 Выведем имена всех файлов из file.txt, имеющих расширение .conf, после чего запишем их в новый текстовый файл conf.txt.

```
[akbeljchug@vbox ~]$ grep .conf file.txt > conf.txt
[akbeljchug@vbox ~]$ cat conf.txt
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
locale.conf
logrotate.conf
makedumpfile.conf.sample
man_db.conf
mke2fs.conf
mtools.conf
ndctl.conf.d
netconfig
nfs.conf
nfsmount.conf
nilfs_cleaner.conf
nsswitch.conf
opensc.conf
opensc-x86_64.conf
passim.conf
passwdqc.conf
pkgconfig
reader.conf.d
request-key.conf
resolv.conf
rsyncd.conf
rsyslog.conf
sddm.conf
sddm.conf.d
```

Рис. 2.2: Поиск расширения .conf

4 Определили, какие файлы в нашем домашнем каталоге имеют имена, начинавшиеся с символа с?

```
[akbeljchug@vbox ~]$ ls ~/c*
/home/akbeljchug/conf.txt
[akbeljchug@vbox ~]$
```

Рис. 2.3: Поиск файлов

5 Выведем на экран (постранично) имена файлов из каталога /etc, начинающиеся с символа h.

```
find /etc -name "h*" -print | less
```



```

find: '/etc/audit': Отказано в доступе
find: '/etc/credstore.encrypted': Отказано в доступе
find: /etc/avahi/hosts
'/etc/credstore': Отказано в доступе
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
find: '/etc/ipsec.d': Отказано в доступе
find: '/etc/lvm/archive': Отказано в доступе
/etc/libibverbs.d/hfi1verbs.driver
/etc/libibverbs.d/hns.driver
find: '/etc/lvm/backup': Отказано в доступе
find: '/etc/lvm/cache': Отказано в доступе
find: '/etc/lvm/devices': Отказано в доступе
find: '/etc/nftables': Отказано в доступе
find: '/etc/openvpn/client': Отказано в доступе
find: '/etc/openvpn/server': Отказано в доступе
/etc/nvme/hostnqn
/etc/nvme/hostid
find: '/etc/pki/rsyslog': Отказано в доступе
find: '/etc/polkit-1/localauthority': Отказано в доступе
find: '/etc/polkit-1/rules.d': Отказано в доступе
find: '/etc/sos/cleaner': Отказано в доступе
find: '/etc/ssh/ssh_config.d': Отказано в доступе
find: '/etc/ssh/sshd_config.d': Отказано в доступе
find: '/etc/ssh/sssd': Отказано в доступе
find: '/etc/sudoers.d': Отказано в доступе
/etc/udev/hwdb.d
/etc/udev/hwdb.bin
/etc/host.conf
/etc/hosts
/etc/hostname

```

Рис. 2.4: Поиск файлов

- 6 Запустили в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. Процесс выполнен
- 7 Удалили файл ~/logfile. Но сначала убили процесс в нем.

```

[akbeljchug@vbox ~]$ find ~ -name "log*" > logfile &
[7] 1755
[akbeljchug@vbox ~]$ rm logfile
[7]   Завершён      find ~ -name "log*" > logfile
[akbeljchug@vbox ~]$

```

Рис. 2.5: Фоновый запуск процесса

- 8 Запустили из консоли в фоновом режиме редактор gedit.
- 9 Определили идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep
- 10 Прочитали справку (man) команды kill, после чего используйте её для за-

вершения процесса gedit.

```
[akbeljchug@vbox ~]$ gedit &
[7] 2782
[akbeljchug@vbox ~]$ ps | grep gedit
2782 pts/0    00:00:00 gedit
[akbeljchug@vbox ~]$ kill 2782
[akbeljchug@vbox ~]$
```

Рис. 2.6: Фоновый запуск и завершение процесса

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

```
df(1) User Commands
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. If no FILE is given, the space available on all currently mounted file systems is shown. Space is shown in 1K blocks by default. 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 the file system containing the device node. This version of df cannot show the space available on unmounted file systems, because it requires non-portable intimate knowledge of file system structures.
OPTIONS
  Show information about the file system on which each FILE resides, or all file systems by default.

  Mandatory arguments to long options are mandatory for short options too.
  -a, --all
      include pseudo, duplicate, inaccessible file systems
  -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
  --direct
      show statistics for a file instead of mount point
  -H, --human-readable
      print sizes in powers of 1024 (e.g., 1023M)
  -H, --si
      print sizes in powers of 1000 (e.g., 1.1G)
  -l, --inodes
      list inode information instead of block usage
  -k, --block-size=K
      like --block-size=1K
  -l, --local
```

Рис. 2.7: Справка по команде `df`

```
du(1) User Commands
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.
  -a, --all
      one each output line with NA, 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
  -b, --bytes
      equivalent to '--apparent-size --block-size=1'
  -c, --total
      produce a grand total
  -d, --dereference-args
      dereference only symlinks that are listed on the command line
  -d, --max-depth=N
      print the total for a directory (or file, with --all) only if it is N or fewer levels below the command line argument; --max-depth=0 is the same as --summarize
  --files0-from=F
      summarize device usage of the NA-terminated file names specified in file F; if F is -, then read names from standard input
Manual page du(1) line 1 (press h for help or q to quit)
```

Рис. 2.8: Запуск команды `df`

```
(akbeljchug@vbox ~)$ df
```

Файловая система	1K-блоков	Использовано	Доступно	Использован%	Смонтировано в
/dev/sda3	82221856	8807892	72278812	11%	/
devtmpfs	4096	0	4096	0%	/dev
tmpfs	4551992	3400	4548592	1%	/dev/shm
efivarfs	256	8	244	3%	/sys/firmware/efi/efivars
tmpfs	1820800	1208	1819592	1%	/run
tmpfs	1024	0	1024	0%	/run/credentials/systemd-journald.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-network-generator.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-udev-load-credentials.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-sysctl.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-tmpfiles-setup-dev-early.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-tmpfiles-setup-dev.service
tmpfs	4551992	32	4551964	1%	/tmp
/dev/sda3	82221856	8807892	72278812	11%	/home
tmpfs	1024	0	1024	0%	/run/credentials/systemd-vconsole-setup.service
/dev/sda2	996780	300204	627764	33%	/boot
/dev/sda1	613168	19796	593364	4%	/boot/efi
tmpfs	1024	0	1024	0%	/run/credentials/systemd-tmpfiles-setup.service
tmpfs	1024	0	1024	0%	/run/credentials/systemd-resolved.service
1234	499065852	164878064	334187788	34%	/media/sf_1234
tmpfs	910396	92	910304	1%	/run/user/1000

```
(akbeljchug@vbox ~)$
```

Рис. 2.9: Справка по команде du

```
12      ./git-test/.git/refs/remotes/origin
12      ./git-test/.git/refs/remotes
24      ./git-test/.git/refs
0       ./git-test/.git/objects/pack
0       ./git-test/.git/objects/info
4       ./git-test/.git/objects/8c
4       ./git-test/.git/objects/6f
4       ./git-test/.git/objects/63
4       ./git-test/.git/objects/fa
4       ./git-test/.git/objects/be
4       ./git-test/.git/objects/f6
4       ./git-test/.git/objects/c4
4       ./git-test/.git/objects/24
4       ./git-test/.git/objects/21
4       ./git-test/.git/objects/60
4       ./git-test/.git/objects/3c
4       ./git-test/.git/objects/1e
4       ./git-test/.git/objects/40
4       ./git-test/.git/objects/1d
4       ./git-test/.git/objects/12
60      ./git-test/.git/objects
4       ./git-test/.git/logs/refs/heads/release
12      ./git-test/.git/logs/refs/heads
4       ./git-test/.git/logs/refs/remotes/origin/release
12      ./git-test/.git/logs/refs/remotes/origin
12      ./git-test/.git/logs/refs/remotes
24      ./git-test/.git/logs/refs
28      ./git-test/.git/logs
204     ./git-test/.git
216     ./git-test
0       ./monthly
0       ./reports/monthly/monthly
0       ./reports/monthly
0       ./reports
12      ./ski.places/equipment
0       ./ski.places/ski.places/plans
0       ./ski.places/ski.places
0       ./ski.places/australia
0       ./ski.places/play/games/play
0       ./ski.places/play/games
0       ./ski.places/play
12      ./ski.places
561732 .
```

Рис. 2.10: Запуск команды du

12 Воспользовавшись справкой команды find, вывести имена всех директорий,

имеющихся в нашем домашнем каталоге.

```
find ~ -type d
```

```
/home/akbeljchug/git-test/.git/logs/refs/remotes/origin
/home/akbeljchug/git-test/.git/logs/refs/remotes/origin/main
/home/akbeljchug/git-test/.git/logs/refs/remotes/origin/develop
/home/akbeljchug/git-test/.git/logs/refs/remotes/origin/release
/home/akbeljchug/git-test/.git/logs/refs/remotes/origin/release/1.0.0
/home/akbeljchug/git-test/.git/HEAD
/home/akbeljchug/git-test/.git/ORIG_HEAD
/home/akbeljchug/git-test/.git/config
/home/akbeljchug/git-test/.git/index
/home/akbeljchug/git-test/README.md
/home/akbeljchug/git-test/package.json
/home/akbeljchug/git-test/CHANGELOG.md
/home/akbeljchug/abc1
/home/akbeljchug/may
/home/akbeljchug/monthly
/home/akbeljchug/monthly/april
/home/akbeljchug/monthly/may
/home/akbeljchug/monthly/june
/home/akbeljchug/reports
/home/akbeljchug/reports/monthly
/home/akbeljchug/reports/monthly/monthly
/home/akbeljchug/reports/monthly/monthly/april
/home/akbeljchug/reports/monthly/monthly/may
/home/akbeljchug/reports/monthly/monthly/june
/home/akbeljchug/reports/monthly/july
/home/akbeljchug/ski.places
/home/akbeljchug/ski.places/equipment
/home/akbeljchug/ski.places/equipment/equiplist
/home/akbeljchug/ski.places/equipment/equiplist2
/home/akbeljchug/ski.places/ski.places
/home/akbeljchug/ski.places/ski.places/plans
/home/akbeljchug/ski.places/australia
/home/akbeljchug/ski.places/play
/home/akbeljchug/ski.places/play/file.old
/home/akbeljchug/ski.places/play/games
/home/akbeljchug/ski.places/play/games/play
/home/akbeljchug/ski.places/play/games/play/file.old
/home/akbeljchug/ski.places/my_os
/home/akbeljchug/ski.places/feathers
/home/akbeljchug/file.txt
/home/akbeljchug/conf.txt
find: 'type': Нет такого файла или каталога
find: 'd': Нет такого файла или каталога
```

Рис. 2.11: Поиск директорий

3 Вывод

В данной работе мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. А также приобрели практические навыки по управлению процессами.

4 Контрольные вопросы

1. Какие потоки ввода вывода вы знаете? Ответ:
 - a) `stdin` — стандартный поток ввода (клавиатура),
 - b) `stdout` — стандартный поток вывода (консоль),
 - c) `stderr` — стандартный поток вывод сообщений об ошибках на экран
2. Объясните разницу между операцией `>` и `>>` Ответ: Разница заключается в том, что Символ `>` используется для переназначения стандартного ввода команды, а символ `>>` используется для присоединения данных в конец файла стандартного вывода команды.
3. Что такое конвейер? Ответ: Конвейер – это способ связи между двумя программами. Например: конвейер `pipe` служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передается последующей. Синтаксис у конвейера следующий:
`команда1 | команда 2`
4. Что такое процесс? Чем это понятие отличается от программы? Ответ: Процесс - это программа, которая выполняется в отдельном виртуальном адресном пространстве независимо от других программ или их пользованию по необходимости.

5. Что такое PID и GID? Ответ: Во первых id — UNIX-утилита, выводящая информацию об указанном пользователе USERNAME или текущем пользователе, который запустил данную команду и не указал явно имя пользователя.
- 1) GID – (Group ID) - идентификатор группы
- 2) UID – (User ID) - идентификатор группы Обычно UID является — положительным целым числом в диапазоне от 0 до 65535, по которому в системе однозначно отслеживаются действия пользователя
6. Что такое задачи и какая команда позволяет ими управлять? Ответ: Запущенные фоновые программы называются задачами(процессами) (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент процессов. Для завершения процесса необходимо выполнить команду : kill % номер задачи
7. Найдите информацию об утилитах top и htop. Каковы их функции? Ответ: Top это консольная команда, которая выводит список работающих в системе процессов и информации о них. По умолчанию она в реальном времени сортирует их по нагрузке на процессор. Htop же является альтернативой программе top она предназначена для вывода на терминал списка запущенных процессов и информации о них.
8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды. Ответ: Команда find используется для поиска и отображения имен файлов, соответствующих заданной строке символов. Синтаксис: find trek [-options] Пример: Задача - Вывести на экран имена файлов из каталога /etc и его подкаталогов, Заканчивающихся на k:
find ~ -name "*k" -print
9. Можно ли по контексту (содержанию) найти файл? Если да, то как? Ответ: Можно, команда ggrep способна обрабатывать вывод других файлов. Для этого надо использовать конвейер, связав вывод команды с вводом ggrep.

Пример: Задача - показать строки в каталоге /dreams с именами начинающимися на t, в которых есть фраза: I like of Operating systems grep I like of Operating systems t*

10. Как определить объем свободной памяти на жёстком диске? Ответ: Команда df показывает размер каждого смонтированного раздела диска. Например команда: df -h
11. Как определить объем вашего домашнего каталога? Ответ: Команда du показывает число килобайт, используемое каждым файлом или каталогом. Например команда: du -sh
12. Как удалить зависший процесс? Ответ: Перед тем, как выполнить остановку процесса, нужно определить его PID. Когда известен PID , мы можем убить его командой kill. Команда kill принимает в качестве параметра PID процесса. PID можно узнать с помощью команд ps, grep, top или htop