

Лабораторная работа №6

Поиск файлов. Перенаправление ввода-вывода. Просмотр запущенных процессов

Старовойтов Егор Сергеевич

Содержание

Цель работы	2
Задание	2
Теоретическое введение	3
Перенаправление ввода-вывода	3
Конвейер	3
Поиск файла	4
Фильтрация текста	4
Проверка использования диска	5
Управление задачами	5
Управление процессами	5
Получение информации о процессах	5
Выполнение лабораторной работы	7
Шаг 1 - Вход в систему	7
Шаг 2	7
Шаг 3	8
Шаг 4	10
Шаг 5	11
Шаг 6	11
Шаг 7	11
Шаг 8	11
Шаг 9	12
Шаг 10	12
Шаг 11	13
Шаг 12	14
Вывод	15
Контрольные вопросы	15
1. Какие потоки ввода вывода вы знаете?	15

2. Объясните разницу между операцией > и >>.....	15
3. Что такое конвейер?.....	15
4. Что такое процесс? Чем это понятие отличается от программы?	15
5. Что такое PID и GID?	16
6. Что такое задачи и какая команда позволяет ими управлять?.....	16
7. Найдите информацию об утилитах top и htop. Каковы их функции?	16
8. Назовите и дайте характеристику команде поиска файлов. Приведите примеры использования этой команды.	16
9. Можно ли по контексту (содержанию) найти файл? Если да, то как?	17
10. Как определить объем свободной памяти на жёстком диске?	17
11. Как определить объем вашего домашнего каталога?	17
12. Как удалить зависший процесс?	17

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.

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, начинавшиеся с символа с? 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, имеющих в вашем домашнем каталоге.

Теоретическое введение

Перенаправление ввода-вывода

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.

Большинство используемых в консоли команд и программ записывают результаты своей работы в стандартный поток вывода `stdout`. Например, команда `ls` выводит в стандартный поток вывода (консоль) список файлов в текущей директории. Потоки вывода и ввода можно перенаправлять на другие файлы или устройства. Проще всего это делается с помощью символов `>`, `>>`, `<`, `<<`. Рассмотрим пример.

```
# Перенаправление stdout (вывода) в файл.  
# Если файл отсутствовал, то он создаётся,  
# иначе -- перезаписывается.
```

```
# Создаёт файл, содержащий список дерева каталогов.  
ls -lR > dir-tree.list
```

```
1>filename  
# Перенаправление вывода (stdout) в файл "filename".  
1>>filename  
# Перенаправление вывода (stdout) в файл "filename",  
# файл открывается в режиме добавления.  
2>filename  
# Перенаправление stderr в файл "filename".  
2>>filename  
# Перенаправление stderr в файл "filename",  
# файл открывается в режиме добавления.  
&>filename  
# Перенаправление stdout и stderr в файл "filename".
```

Конвейер

Конвейер (`pipe`) служит для объединения простых команд или утилит в цепочки, в которых результат работы предыдущей команды передаётся последующей. Синтаксис следующий:

```
команда 1 | команда 2  
# означает, что вывод команды 1 передаётся на ввод команде 2
```

Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например:

```
1 ls -la |sort > sortilg_list
```

вывод команды `ls -la` передаётся команде сортировки `sort\verb`, которая пишет результат в файл `sorting_list\verb`.

Чаще всего скрипты на Bash используются в качестве автоматизации каких-то рутинных операций в консоли, отсюда иногда возникает необходимость в обработке `stdout` одной команды и передача на `stdin` другой команде, при этом результат выполнения команды должен обработан.

Поиск файла

Команда **find** используется для поиска и отображения на экран имён файлов, соответствующих заданной строке символов.

Формат команды: `find путь [-опции]`

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Примеры:

1. Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на `f`:

```
find ~ -name "f*" -print
```

Здесь `~` — обозначение вашего домашнего каталога, `-name` — после этой опции указывается имя файла, который нужно найти, `"f*"` — строка символов, определяющая имя файла, `-print` — опция, задающая вывод результатов поиска на экран.

2. Вывести на экран имена файлов в каталоге `/etc`, начинающихся с символа `p`:

```
find /etc -name "p*" -print
```

3. Найти в Вашем домашнем каталоге файлы, имена которых заканчиваются символом

и удалить их:

```
find ~ -name "*~" -exec rm "{}" \;
```

Здесь опция `-exec rm "{}"`; задаёт применение команды `rm` ко всем файлам, имена которых соответствуют указанной после опции `-name` строке символов. Для просмотра опций команды `find` воспользуйтесь командой `man`.

Фильтрация текста

Найти в текстовом файле указанную строку символов позволяет команда **grep**. Формат команды: `grep строка имя_файла` Кроме того, команда `grep` способна обрабатывать стандартный вывод других команд (любой текст). Для этого следует использовать конвейер, связав вывод команды с вводом `grep`.

Примеры:

1. Показать строки во всех файлах в вашем домашнем каталоге с именами, начинающимися на f, в которых есть слово begin:

```
grep begin f*
```

2. Найти в текущем каталоге все файлы, содержащих в имени «лаб»:

```
ls -l | grep лаб
```

Проверка использования диска

- Команда **df** показывает размер каждого смонтированного раздела диска. Формат команды: `df [-опции] [файловая_система]`

Пример:

```
df -vi
```

- Команда **du** показывает число килобайт, используемое каждым файлом или каталогом. Формат команды: `du [-опции] [имя_файла...]`

Пример.

```
du -a ~/
```

Управление задачами

Любую выполняющуюся в консоли команду или внешнюю программу можно запустить в фоновом режиме. Для этого следует в конце имени команды указать знак амперсанда &.

Например:

```
gedit &
```

Будет запущен текстовый редактор gedit в фоновом режиме. Консоль при этом не будет заблокирована.

Запущенные фоном программы называются задачами (jobs). Ими можно управлять с помощью команды jobs, которая выводит список запущенных в данный момент задач.

Для завершения задачи необходимо выполнить команду `kill %номер задачи`

Управление процессами

Любой команде, выполняемой в системе, присваивается идентификатор процесса (process ID). Получить информацию о процессе и управлять им, пользуясь идентификатором процесса, можно из любого окна командного интерпретатора.

Получение информации о процессах

Команда **ps** используется для получения информации о процессах. Формат команды: `ps [-опции]`

Для получения информации о процессах, управляемых вами и запущенных (работающих или остановленных) на вашем терминале, используйте опцию **aux**.

Пример:

```
ps aux
```

Для запуска команды в фоновом режиме необходимо в конце командной строки указать знак & (амперсанд).

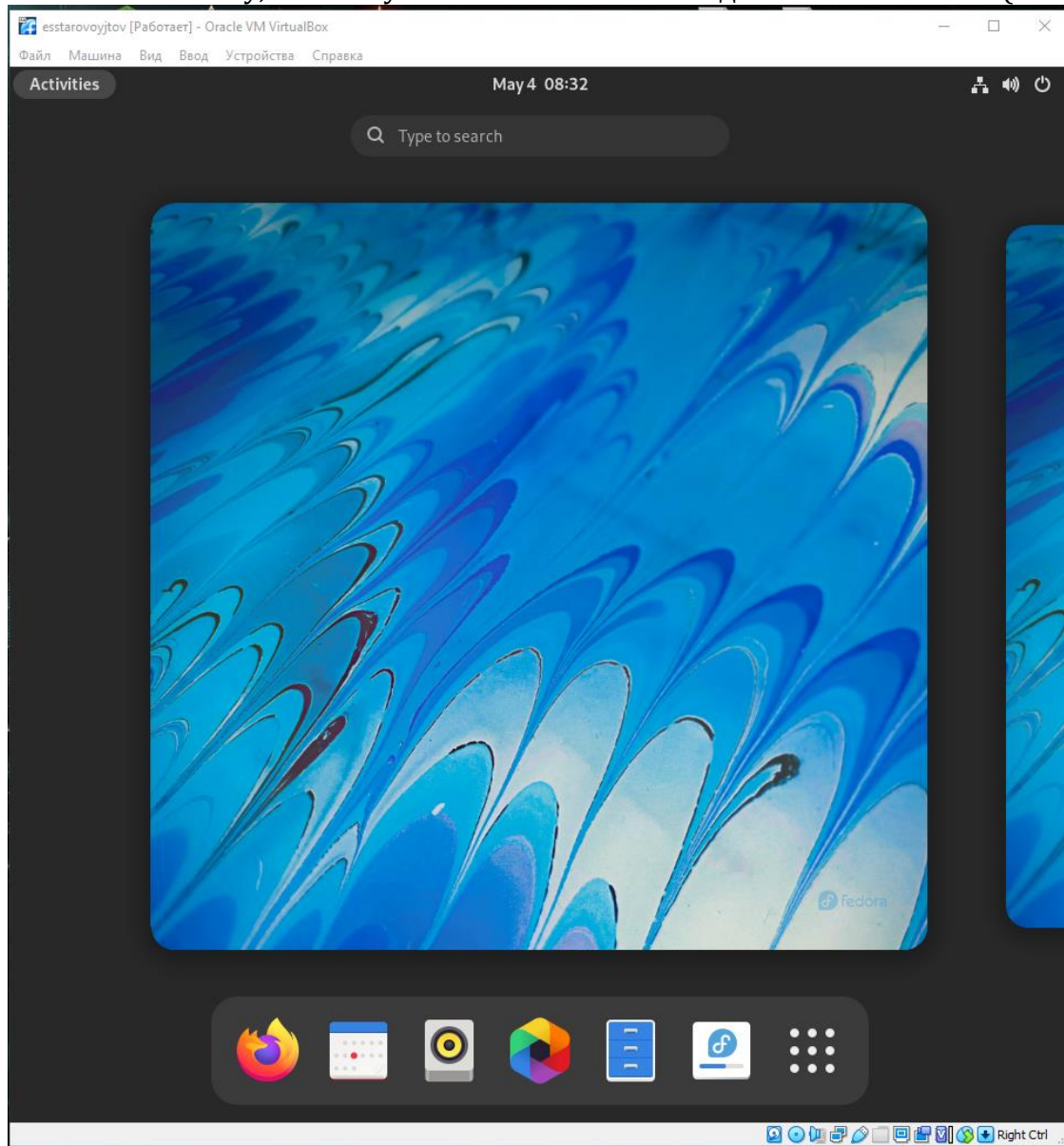
Пример работы, требующей много машинного времени для выполнения, и которую целесообразно запустить в фоновом режиме:

```
find /var/log -name "*.log" -print > l.log &
```

Выполнение лабораторной работы

Шаг 1 - Вход в систему

Я зашел в систему, используя имя пользователя из дисплейного класса (esstarovoyjtov).



Шаг 2

Я записал в файл "file.txt" названия файлов, содержащихся в каталоге /etc. Использовал команду find. Чтобы узнать содержимое каталога только первого уровня вложенности нужно указать опциональный аргумент -maxdepth со значением 1, чтобы искать только файлы, нужно указать опцию -type со значением f. Вывод команды я направил в "file.txt" с помощью оператора ">". Результат выполнения команды я просмотрел с помощью cat.

```
[liveuser@localhost-live ~]$ find /etc -maxdepth 1 -type f > file.txt
[liveuser@localhost-live ~]$ cat file.txt
/etc/protocols
/etc/hostname
/etc/subuid
/etc/vconsole.conf
/etc/rpc
/etc/fstab
/etc/request-key.conf
/etc/magic
/etc/swtpm_setup.conf
/etc/bindresvport.blacklist
/etc/trusted-key.key
/etc/passwd
```

Название файлов из /etc

Затем в этот же файл я записал имена всех файлов, содержащихся в домашнем каталоге. Чтобы при записи в “file.txt” не потерять предыдущие данные, я указал оператор “>>” (дозапись).

```
[liveuser@localhost-live ~]$ find ~ -maxdepth 1 -type f >> file.txt
[liveuser@localhost-live ~]$ cat file.txt
```

Дозапись в “file.txt”

```
/etc/libuser.conf
/etc/exports
/etc/sudoers
/etc/csh.login
/etc/dracut.conf
/etc/idmapd.conf
/etc/passwd-
/etc/usb_modeswitch.conf
/etc/DIR_COLORS.lightbgcolor
/etc/DIR_COLORS
/etc/krb5.conf
/etc/services
/etc/subgid
/etc/shadow
/etc/host.conf
/etc/updatedb.conf
/etc/brlapi.key
/etc/mime.types
/etc/ethertypes
/etc/sysctl.conf
/etc/hosts
/etc/networks
/etc/dleyna-server-service.conf
/etc/nsswitch.conf.bak
/etc/.updated
/home/liveuser/.vboxclient-seamless.pid
/home/liveuser/.bash_logout
/home/liveuser/.vboxclient-clipboard.pid
/home/liveuser/.bash_profile
/home/liveuser/.vboxclient-draganddrop.pid
/home/liveuser/.bash_history
/home/liveuser/.bashrc
/home/liveuser/file.txt
[liveuser@localhost-live ~]$
```

Содержимое “file.txt”

Шаг 3

С помощью утилиты grep я вывел имена всех файлов из “file.txt”, имеющих расширение .conf. Аргументом я передал подстроку, которую надо найти в именах файлов. grep использует синтаксис регулярных выражений, поэтому я указал после .conf символ \$ (конец строки), а перед точкой символ экранирования.


```
/etc/nsswitch.conf.bak
[liveuser@localhost-live ~]$ grep "\.conf$" file.txt
/etc/vconsole.conf
/etc/request-key.conf
/etc/swtpm_setup.conf
/etc/chrony.conf
/etc/sestatus.conf
/etc/asound.conf
/etc/appstream.conf
/etc/ld.so.conf
/etc/nfs.conf
/etc/xattr.conf
/etc/passwdqc.conf
/etc/dnsmasq.conf
/etc/brltty.conf
/etc/radvd.conf
/etc/nfsmount.conf
/etc/rygel.conf
/etc/jwhois.conf
/etc/kdump.conf
/etc/nsswitch.conf
/etc/anthy-unicode.conf
/etc/swtpm-localca.conf
/etc/tcsd.conf
/etc/man_db.conf
/etc/mtools.conf
/etc/locale.conf
/etc/rsyncd.conf
/etc/libaudit.conf
/etc/dley-na-renderer-service.conf
/etc/uresourced.conf
/etc/logrotate.conf
/etc/mke2fs.conf
/etc/opensc.conf
/etc/fuse.conf
/etc/fprintd.conf
/etc/sudo.conf
/etc/Trolltech.conf
/etc/libuser.conf
/etc/dracut.conf
/etc/idmapd.conf
/etc/usb_modeswitch.conf
/etc/krb5.conf
/etc/host.conf
/etc/updatedb.conf
/etc/sysctl.conf
/etc/dley-na-server-service.conf
[liveuser@localhost-live ~]$
```

Список файлов с расширением .conf

Также я записал все найденные названия файлов в "conf.txt", перенаправив вывод предыдущей команды в файл.

```
[liveuser@localhost-live ~]$ grep "\.conf$" file.txt >> conf.txt
[liveuser@localhost-live ~]$ cat conf.txt
/etc/vconsole.conf
/etc/request-key.conf
/etc/swtpm_setup.conf
/etc/chrony.conf
/etc/sestatus.conf
/etc/asound.conf
/etc/appstream.conf
/etc/ld.so.conf
/etc/nfs.conf
/etc/xattr.conf
/etc/passwdqc.conf
/etc/dnsmasq.conf
/etc/brltty.conf
/etc/radvd.conf
/etc/nfsmount.conf
/etc/rygel.conf
/etc/jwhois.conf
/etc/kdump.conf
/etc/nsswitch.conf
/etc/anthy-unicode.conf
/etc/swtpm-localca.conf
/etc/tcsd.conf
/etc/man_db.conf
/etc/mtools.conf
/etc/locale.conf
/etc/rsyncd.conf
/etc/libaudit.conf
/etc/dley-na-renderer-service.conf
/etc/uresourced.conf
/etc/logrotate.conf
/etc/mke2fs.conf
/etc/opensc.conf
/etc/fuse.conf
/etc/fprintd.conf
/etc/sudo.conf
/etc/Trolltech.conf
/etc/libuser.conf
/etc/dracut.conf
/etc/ldapd.conf
/etc/usb_modeswitch.conf
/etc/krb5.conf
/etc/host.conf
/etc/updatedb.conf
/etc/sysctl.conf
/etc/dley-na-server-service.conf
[liveuser@localhost-live ~]$
```

Запись списка файлов с расширением .conf в ".conf.txt"

Шаг 4

Я определил двумя способами (find и grep) имена файлов домашнего каталога, которые начинаются с символа "с". Такой файл всего один - "conf.txt". Чтобы узнать содержимое каталога только первого уровня вложенности нужно указать опциональный аргумент -

maxdepth со значением 1, чтобы искать только файлы, нужно указать опцию -type со значением f.

```
[liveuser@localhost-live ~]$ find ~ -maxdepth 1 -type f -name "c*" -print
/home/liveuser/conf.txt
[liveuser@localhost-live ~]$ ls | grep -e "^c*"
conf.txt
```

Файлы в ~ на букву "с"

Шаг 5

Я вывел на экран имена файлов из каталога /etc, начинающихся с символа h.

Чтобы узнать содержимое каталога только первого уровня вложенности нужно указать опциональный аргумент -maxdepth со значением 1, чтобы искать только файлы, нужно указать опцию -type со значением f.

```
[liveuser@localhost-live ~]$ find /etc -maxdepth 1 -type f -name "h*" -print
/etc/hostname
/etc/host.conf
/etc/hosts
[liveuser@localhost-live ~]$
```

Файлы в /etc на букву "h"

Шаг 6

Я запустил в фоновом режиме процесс, записывающий в файл ~/logfile имена файлов, название которых начинается с "log". Поиск осуществляется с помощью команды find.

```
[liveuser@localhost-live ~]$ find ~ -name "log*" -print > logfile &
[1] 4082
[1]+  Done                  find ~ -name "log*" -print > logfile
[liveuser@localhost-live ~]$ cat logfile
/home/liveuser/logfile
[liveuser@localhost-live ~]$
```

Запуск фонового процесса

Шаг 7

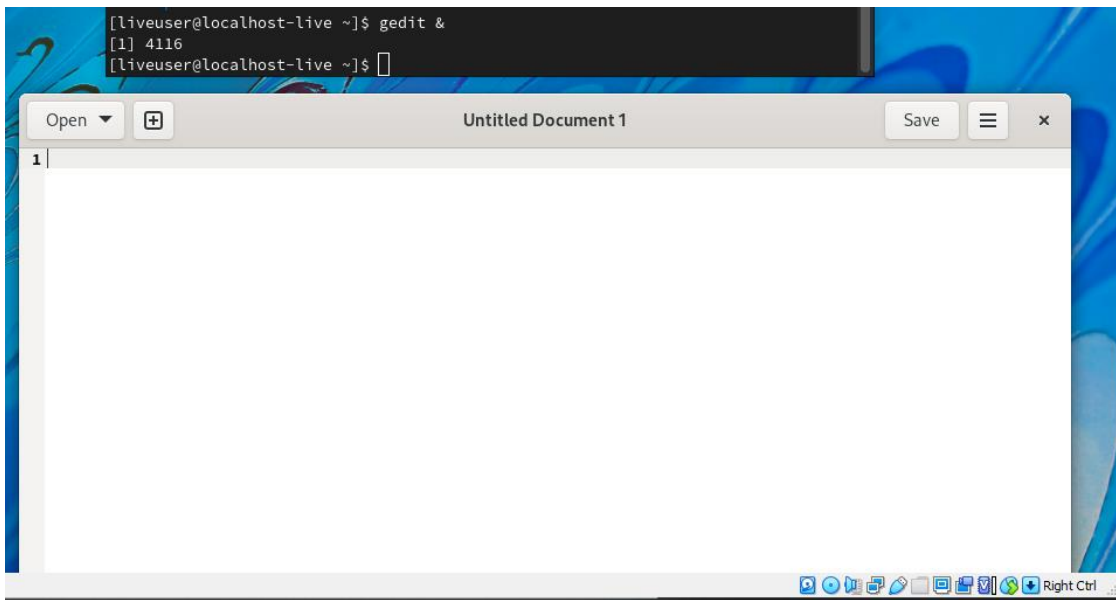
Удалил файл ~/logfile.

```
[liveuser@localhost-live ~]$ rm logfile
[liveuser@localhost-live ~]$ ls
conf.txt  Documents  file.txt  Pictures  Templates
Desktop  Downloads  Music     Public    Videos
[liveuser@localhost-live ~]$
```

Удаление ~/logfile

Шаг 8

Запуск редактора gedit в фоновом режиме. Идентификатор процесса gedit - 4116.



Запуск gedit в фоновом режиме

Шаг 9

Используя команду `ps`, которая выводит список текущих процессов, конвейера и утилиты `grep` я узнал идентификатор запущенного процесса `gedit`. Другой способ узнать PID `gedit` - посмотреть вывод в консоль предыдущего шага.

```
[liveuser@localhost-live ~]$ gedit &
[1] 4116
[liveuser@localhost-live ~]$ ps
  PID TTY          TIME CMD
  4022 pts/0    00:00:00 bash
   4116 pts/0    00:00:01 gedit
   4142 pts/0    00:00:00 ps
[liveuser@localhost-live ~]$ ps | grep -e "gedit"
   4116 pts/0    00:00:01 gedit
[liveuser@localhost-live ~]$
```

Идентификатор gedit

Шаг 10

Я прочитал справку команды **kill** и узнал, как с ее помощью завершить процесс `gedit`. Команды `kill` посылает различные сигналы процессам, для передачи сигнала выключения можно использовать опцию `-KILL`.

```
[liveuser@localhost-live ~]$ man kill
[liveuser@localhost-live ~]$ kill -KILL 4116
[1]+  Killed                  gedit
[liveuser@localhost-live ~]$
```

Завершение процесса gedit

```
KILL(1)                                User Commands                                KILL(1)

NAME
    kill - terminate a process

SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds
    signal] [--] pid|name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified processes
    or process groups.

    If no signal is specified, the TERM signal is sent. The default action
    for this signal is to terminate the process. This signal should be used
    in preference to the KILL signal (number 9), since a process may
    install a handler for the TERM signal in order to perform clean-up
    steps before terminating in an orderly fashion. If a process does not
    terminate after a TERM signal has been sent, then the KILL signal may
    be used; be aware that the latter signal cannot be caught, and so does
    not give the target process the opportunity to perform any clean-up

Manual page kill(1) line 1/160 16% (press h for help or q to quit)
```

Справка про kill

Шар 11

Я получил подробную информацию по командам df и du с помощью man.

```
DU(1)                                User Commands                                DU(1)

NAME
    du - estimate file space usage

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

DESCRIPTION
    Summarize disk 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 disk usage; although the ap-

Manual page du(1) line 1 (press h for help or q to quit)
```

```

[liveuser@localhost-live ~]$ man df
[liveuser@localhost-live ~]$ man du
[liveuser@localhost-live ~]$ df
Filesystem            1K-blocks    Used Available Use% Mounted on
devtmpfs                1970628         0   1970628  0% /dev
tmpfs                   2004136         0   2004136  0% /dev/shm
tmpfs                    801656     9552    792104  2% /run
/dev/sr0                 1962240 1962240         0 100% /run/initramfs/live
/dev/mapper/live-rw     7640588 5502460   2121744  73% /
tmpfs                   2004140         52   2004088  1% /tmp
vartmp                  2004136         0   2004136  0% /var/tmp
tmpfs                    400824     136    400688  1% /run/user/1000
/dev/sr1                  60096     60096         0 100% /run/media/liveuser/VBox_GA
s_6.1.34
[liveuser@localhost-live ~]$ du
4      ./mozilla/extensions
4      ./mozilla/plugins
12     ./mozilla
8      ./config/gtk-3.0
4      ./config/enchant
8      ./config/dconf
16     ./config/evolution/sources
20     ./config/evolution
4      ./config/goa-1.0
8      ./config/autostart
4      ./config/abrt
8      ./config/ibus/bus
12     ./config/ibus
4      ./config/nautilus
4      ./config/gnome-boxes/sources
8      ./config/gnome-boxes
8      ./config/pulse
104    ./config
76     ./local/share/gvfs-metadata
4      ./local/share/icc
4      ./local/share/keyrings
4      ./local/share/flatpak/db
8      ./local/share/flatpak
4      ./local/share/sounds
2932   ./local/share/gnome-photos/tracker3/private
2936   ./local/share/gnome-photos/tracker3
2940   ./local/share/gnome-photos
4      ./local/share/applications
4      ./local/share/foiks
4      ./local/share/evolution/addressbook/system/photos

```

Вывод команд `df` и `du`

Шаг 12

С помощью команды `find` я вывел имена всех директорий, содержащихся в домашнем каталоге. Для поиска именно директорий нужно указать опцию `-type` со значением `d`.

```
[liveuser@localhost-live ~]$ find ~ -maxdepth 1 -type d
/home/liveuser
/home/liveuser/.mozilla
/home/liveuser/.config
/home/liveuser/.local
/home/liveuser/Music
/home/liveuser/Templates
/home/liveuser/Videos
/home/liveuser/.cache
/home/liveuser/Downloads
/home/liveuser/Documents
/home/liveuser/Desktop
/home/liveuser/Public
/home/liveuser/Pictures
[liveuser@localhost-live ~]$
```

Имена директорий в домашнем каталоге

Вывод

Я изучил инструменты поиска файлов и фильтрации текстовых данных, приобрел практические навыки по управлению процессами и заданиями, а также научился получать информацию про диски и файловые системы с помощью команд `du` и `df`.

Контрольные вопросы

1. Какие потоки ввода вывода вы знаете?

- `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 `>>`.

Операция `>` перезаписывает файл вывода новыми данными, а `>>` осуществляет дозапись в конец указанного файла.

3. Что такое конвейер?

Конвейер - это оператор, объединяющий команды в цепочки, в которой результат работы предыдущей команды передается следующей.

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

Программа - это набор инструкций исполнителю, процесс - сущность исполняемой программы. Процесс содержит текст программы, стек ее данных, дескрипторы открытых файлов и другую служебную информацию.

5. Что такое PID и GID?

- PID - это идентификатор процесса, его уникальный номер.
- GID - это идентификатор группы пользователя в Linux.

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

Задачами называются запущенные фоном программы. Узнать список запущенных в данный момент задач можно с помощью команды **jobs**, а посылать им управляющие сигналы можно с помощью команды **kill**.

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

- top (table of processes) - консольная команда, выводящая список работающих в системе процессов и информацию о них. По умолчанию top сортирует все процессы в реальном времени по их нагрузке на процессор.
- htop - Компьютерная программа, предназначенная для вывода на терминал списка запущенных процессов и информации о них. Создана как альтернатива программы top. Написана на языке Си. Предоставляет пользователю текстовый интерфейс, для вывода на терминал использует библиотеку ncurses.

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

Команда **find** используется для поиска и отображения на экран имён файлов, соответствующих заданной строке символов.

Формат команды: `find путь [-опции]`

Путь определяет каталог, начиная с которого по всем подкаталогам будет вестись поиск.

Примеры:

1. Вывести на экран имена файлов из вашего домашнего каталога и его подкаталогов, начинающихся на f:

```
find ~ -name "f*" -print
```

Здесь ~ — обозначение вашего домашнего каталога, -name — после этой опции указывается имя файла, который нужно найти, "f*" — строка символов, определяющая имя файла, -print — опция, задающая вывод результатов поиска на экран.

2. Вывести на экран имена файлов в каталоге /etc, начинающихся с символа p:

```
find /etc -name "p*" -print
```

3. Найти в Вашем домашнем каталоге файлы, имена которых заканчиваются символом

и удалить их:

```
find ~ -name "*~" -exec rm "{}" \;
```

Здесь опция -exec rm "{}" ; задаёт применение команды rm ко всем файлам, имена которых соответствуют указанной после опции -name строке символов.

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

С помощью команды `grep` можно искать файлы по содержимому. Для этого нужно вызвать команду `grep` с опциями `-lir` и искомым текстом, например

```
grep -lir '#include'
```

10. Как определить объем свободной памяти на жёстком диске?

Набрав в консоли

```
df -h
```

можно увидеть объем свободной памяти в каждой файловой системе. Просуммировав эти значения можно узнать объем свободной памяти на жёстком диске.

11. Как определить объем вашего домашнего каталога?

Объем домашнего каталога в удобочитаемом формате можно узнать набрав в терминале

```
du ~ -h
```

12. Как удалить зависший процесс?

Удалить зависший процесс с идентификатором `pid` можно набрав в терминале `kill -KILL pid`.