

Лабораторная работа №7

Дисциплина: Операционные системы

Колчева Юлия Вячеславовна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	12
4	Контрольные вопросы	13

List of Tables

List of Figures

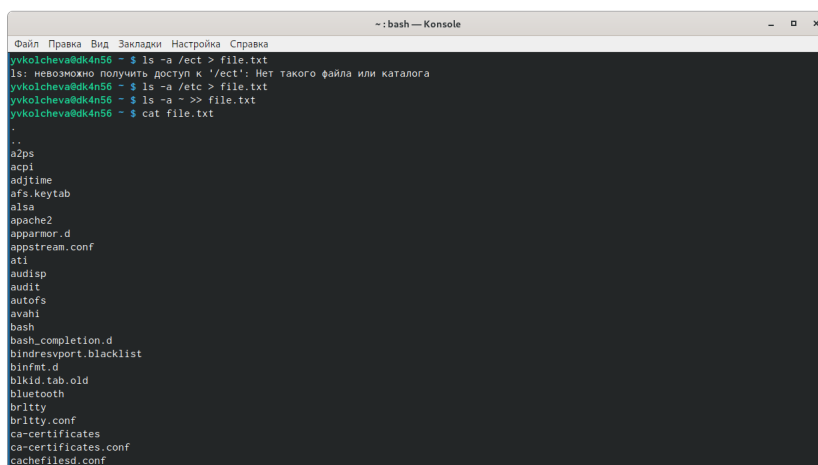
2.1	Перезаписывание	6
2.2	Вывод имён	7
2.3	Слова с символа с	7
2.4	Имена с символа h	7
2.5	Имна с log	8
2.6	Скриншот консоли	8
2.7	Редактор gedit	8
2.8	Команда df	9
2.9	Команда du	9
2.10	Результат	10
2.11	Просмотр команды	10
2.12	Имена	11

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

Осуществляю вход в систему, используя свои логин и пароль. Для того, чтобы записать в файл file.txt названия файлов, содержащихся в каталоге /etc, использую команду «ls-a/etc > file.txt». Далее с помощью команды «ls-a~ » file.txt» дописываю в этот же файл названия файлов, содержащихся в домашнем каталоге. Командой «cat file.txt» просматриваю файл. (рис. 2.1)



```
~: bash — Konsole
Файл Правка Вид Закладки Настройка Справка
yvkolcheva@dk4n56 ~$ ls -a /etc > file.txt
ls: невозможно получить доступ к '/etc': Нет такого файла или каталога
yvkolcheva@dk4n56 ~$ ls -a /etc > file.txt
yvkolcheva@dk4n56 ~$ ls -a ~ >> file.txt
yvkolcheva@dk4n56 ~$ cat file.txt
.
..
a2ps
acpi
adjtime
afs.keytab
alsa
apache2
apparmor.d
appstream.conf
at1
audisp
audit
autofs
avahi
bash
bash_completion.d
bindresvport.blacklist
binfmt.d
blkid.tab.old
bluetooth
brltty
brltty.conf
ca-certificates
ca-certificates.conf
cachefileld.conf
```

Figure 2.1: Перезаписывание

Вывожу имена всех файлов из file.txt, имеющих расширение .conf и записываю их в новый текстовый файл conf.txt с помощью команды «grep-e'.conf\$'file.txt> conf.txt». (рис. 2.2)

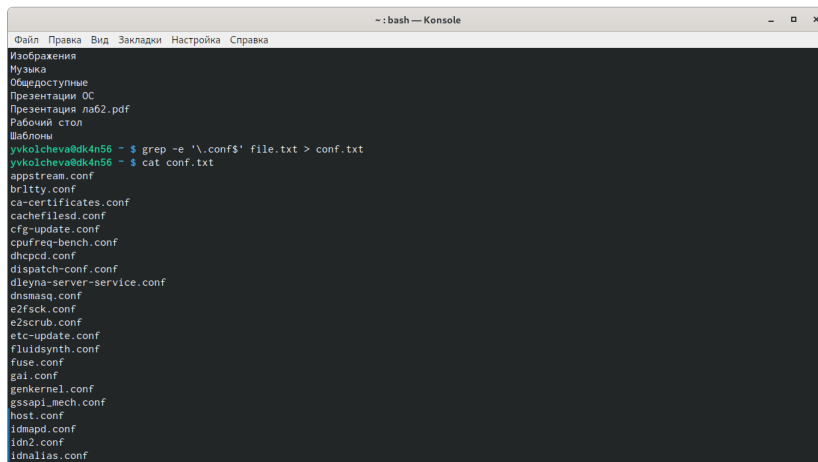


Figure 2.2: Вывод имён

Определяем имена в домашнем каталоге, начинающиеся с символа с, несколькими командами: «find ~ -maxdepth 1 -name "c*" -print», «ls ~/c» и «ls -a ~ | grep c*» (рис. 2.3)

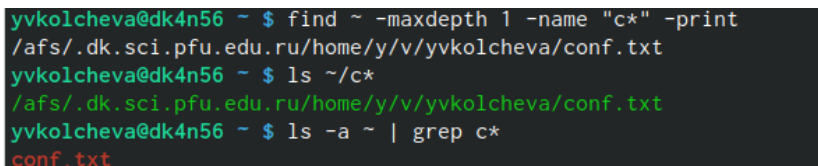


Figure 2.3: Слова с символа с

Чтобы вывести на экран имена файлов из каталога /etc, начинающиеся с символа h, воспользуемся командой «find /etc -maxdepth 1 -name "h*" | less» (рис. 2.4)

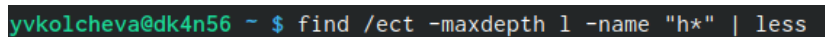


Figure 2.4: Имена с символа h

Запускаю в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log, используя команду «find / -name "log*" > logfile &». Командой «cat logfile» проверяю выполненные действия. Далее удаляю файл ~/logfile командой «rm logfile» (рис. 2.6) (рис. 2.7)

```
yvko1cheva@dk4n56 ~ $ find / -name "log*" > logfile &
[1] 18342
yvko1cheva@dk4n56 ~ $ find: '/mnt/floppy': Отказано в доступе
find: '/mnt/cdrom': Отказано в доступе
```

Figure 2.5: Имна с log

```
yvko1cheva@dk4n56 ~ $ rm logfile
```

Figure 2.6: Скриншот консоли

Запускаю редактор gedit в фоновом режиме командой «gedit &». Чтобы определить идентификатор процесса gedit, использую команду «ps | grep -i "gedit"». Из рисунка видно, что наш процесс имеет номер 19075. Узнать идентификатор процесса можно также, используя команду «pgrep gedit» или «pidof gedit». Прочитав информацию о команде kill с помощью команды «man kill», использую её для завершения процесса gedit. (рис. 2.7)

```

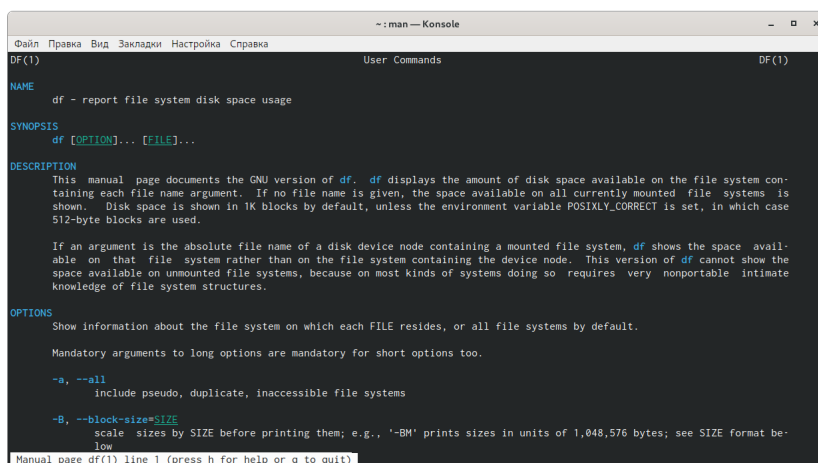
~: bash — Konsole
Файл Правка Вид Закладки Настройка Справка
/usr/portage/app-text/logmerge
/usr/portage/app-text/logmerge/logmerge-0.5.2b.ebuild
/usr/portage/dev-python/logfury
/usr/portage/dev-python/logfury/files/logfury-0.1.2-fix-requirements-remove-dev-yvko1cheva@dk4n56 ~ $ rm logfile
yvkoi1cheva@dk4n56 ~ $ gedit &
[1] 19020
yvkoi1cheva@dk4n56 ~ $ ps | grep -i "gedit"
[1]+  Завершён      gedit
yvkoi1cheva@dk4n56 ~ $ gedit &
[1] 19075
yvkoi1cheva@dk4n56 ~ $ ps | grep -i "gedit"
19075 pts/2    00:00:01 gedit
yvkoi1cheva@dk4n56 ~ $ man kill
yvkoi1cheva@dk4n56 ~ $ kill 19075
yvkoi1cheva@dk4n56 ~ $ man df
[1]+  Завершено      gedit
yvkoi1cheva@dk4n56 ~ $ man du
yvkoi1cheva@dk4n56 ~ $ df
Файловая система      1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
none                    4000212      28404      3971808          1% /run
udev                   3924488          0      3924488          0% /dev
tmpfs                   4000212      53156      3947056          2% /dev/shm
/dev/sda8              491812356    59485100    407321512         13% /
tmpfs                    4096          0         4096          0% /sys/fs/cgroup
tmpfs                   4000216      97352      3902864          3% /tmp
/dev/sda6              91557952     338956     86545052          1% /var/cache/openafs
mark.sci.pfu.edu.ru:/usr/portage 8388608     4353792     1804032         71% /usr/portage
mark.sci.pfu.edu.ru:/usr/local/share/portage 8388608     4353792     1804032         71% /usr/local/share/portage
mark.sci.pfu.edu.ru:/com/lib/portage 733747200    115401984    618345216        16% /com/lib/portage
AFS                    2147483647          0      2147483647          0% /afs
tmpfs                   800040         256       799784          1% /run/user/3712

```

Figure 2.7: Редактор gedit

С помощью команд «man df» и «man du» узнаю информацию по необходимым командам и далее использую их. df – утилита, показывающая список всех файловых систем по именам устройств, сообщает их размер, занятое и свободное пространство и точки монтирования. Синтаксис: df [опции] устройство du – утилита,

предназначенная для вывода информации об объеме дискового пространства, занятого файлами и директориями. Она принимает путь к элементу файловой системы и выводит информацию о количестве байт дискового пространства или блоков диска, задействованных для его хранения. (рис. 2.8) (рис. 2.9) (рис. 2.10)



```
~:man -- Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
DF(1)                                     User Commands                                     DF(1)

NAME
    df - report file system disk space usage

SYNOPSIS
    df [OPTION]... [FILE]...

DESCRIPTION
    This manual page documents the GNU version of df.  df displays the amount of disk 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.  Disk 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 disk 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.

OPTIONS
    Show information about the file system on which each FILE resides, or all file systems by default.

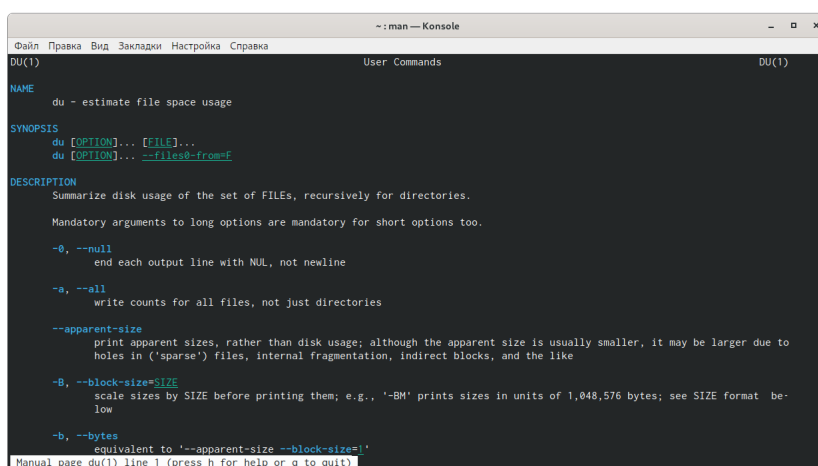
    Mandatory arguments to long options are mandatory for short options too.

    -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

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

Figure 2.8: Команда df



```
~:man -- Konsole
Файл  Правка  Вид  Закладки  Настройка  Справка
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 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'

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

Figure 2.9: Команда du

```
~: bash — Konsole
[1]* Завершено gedit
yvkolcheva@dk4n56 ~$ man du
yvkolcheva@dk4n56 ~$ df
Файловая система      1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
none                  4000212      28404      3971808         1% /run
udev                 3924488         0      3924488         0% /dev
tmpfs                 4000212      53156      3947056         2% /dev/shm
/dev/sda8             491812356    59485100    407321512        13% /
tmpfs                  4096         0         4096         0% /sys/fs/cgroup
tmpfs                 4000216      97352      3902864         3% /tmp
/dev/sda6             91557952     338956     86545052         1% /var/cache/openafs
mark.sci.pfu.edu.ru:/usr/portage 8388608     4253792     1804032        71% /usr/portage
mark.sci.pfu.edu.ru:/usr/local/share/portage 8388608     4253792     1804032        71% /usr/local/share/portage
mark.sci.pfu.edu.ru:/com/lib/portage 733747200    115401884    618345216        16% /com/lib/portage
AFS                   2147483647         0    2147483647         0% /afs
tmpfs                  800040         256      799784         1% /run/user/3712
yvkolcheva@dk4n56 ~$ du
2      ./public/public_html
4      ./public
2      ./config/gnome-session/saved-session
4      ./config/gnome-session
3      ./config/ibus/bus
5      ./config/ibus
9      ./config/occonf
5      ./config/evolution/sources
7      ./config/evolution
3      ./config/goa-1.0
74     ./config/libaccounts-glib
2      ./config/gconf
4      ./config/gtk-3.0
64     ./config/session
```

Figure 2.10: Результат

Вывожу имена всех директорий, имеющихся в моем домашнем каталоге с помощью команды «find~ -typed», получив информацию с помощью команды «man find» (рис. 2.11) (рис. 2.12)

```
~: man — Konsole
FIND(1)                                     General Commands Manual                                     FIND(1)
NAME
  find - search for files in a directory hierarchy
SYNOPSIS
  find [-H] [-L] [-P] [-D debugopts] [-Olevel] [starting-point...] [expression]
DESCRIPTION
  This manual page documents the GNU version of find. GNU find searches the directory tree rooted at each given starting-point by evaluating the given expression from left to right, according to the rules of precedence (see section OPERATORS), until the outcome is known (the left hand side is false for and operations, true for or), at which point find moves on to the next file name. If no starting-point is specified, '.' is assumed.
  If you are using find in an environment where security is important (for example if you are using it to search directories that are writable by other users), you should read the 'Security Considerations' chapter of the findutils documentation, which is called Finding Files and comes with findutils. That document also includes a lot more detail and discussion than this manual page, so you may find it a more useful source of information.
OPTIONS
  The -H, -L and -P options control the treatment of symbolic links. Command-line arguments following these are taken to be names of files or directories to be examined, up to the first argument that begins with '-', or the argument '-' ('' or ''). That argument and any following arguments are taken to be the expression describing what is to be searched for. If no paths are given, the current directory is used. If no expression is given, the expression -print is used (but you should probably consider using -print0 instead, anyway).
  This manual page talks about 'options' within the expression list. These options control the behaviour of find but are specified immediately after the last path name. The five 'real' options -H, -L, -P, -D and -O must appear before the first path name, if at all. A double dash -- can also be used to signal that any remaining arguments are not options (though ensuring that all start points begin with either './' or '/' is generally safer if you use wildcards in the list of start points).
Manual page find(1) line 1 (press h for help or q to quit)
```

Figure 2.11: Просмотр команды

```
~: bash — Konsole
Файл Правка Вид Закладки Настройка Справка
yvkolcheva@dk4n56 ~ % find ~ -type d
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/public
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/public/public_html
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/gnome-session
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/gnome-session/saved-session
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/ibus
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/ibus/bus
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/dconf
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/evolution
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/evolution/sources
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/gss-1.0
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libaccounts-glib
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/gconf
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/gtk-3.0
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/session
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/menubar
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/popupmenu
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/toolbar
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/statusbar
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/images
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/swriter/images/Bitmaps
/afs/.dk.sci.pfu.edu.ru/home/y/v/yvkolcheva/.config/libreoffice/4/user/config/soffice.cfg/modules/schart
```

Figure 2.12: Имена

3 Выводы

В ходе выполнения данной лабораторной работы я изучила инструменты поиска файлов и фильтрации текстовых данных, а также приобрела практические навыки: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

4 Контрольные вопросы

Ответы на контрольные вопросы:) 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.

2) ‘>’ Перенаправление вывода в файл

‘>>’ Перенаправление вывода в файл и открытие файла в режиме добавления (данные добавляются в конец файла)/

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

Синтаксис следующий:

команда1 команда2 (это означает, что вывод команды 1 передастся на ввод команде 2)

- 4) Процесс рассматривается операционной системой как заявка на потребление всех видов ресурсов, кроме одного – процессорного времени. Этот последний важнейший ресурс распределяется операционной системой между другими единицами работы – потоками, которые и получили свое название благодаря тому, что они представляют собой последовательности (потоки выполнения) команд.

Процесс – это выполнение программы. Он считается активной сущностью и реализует действия, указанные в программе.

Программа представляет собой статический набор команд, а процесс это набор ресурсов и данных, использующихся при выполнении программы.

- 5) `pid`: идентификатор процесса (PID) процесса (`processID`), к которому вызывают метод

`gid`: идентификатор группы UNIX, в котором работает программа.

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

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

- 7) `top` – это консольная программа, которая показывает список работающих процессов в системе. Программа в реальном времени отсортирует запущенные процессы по их нагрузке на процессор.

`htop` – это продвинутый консольный мониторинг процессов. Утилита выводит постоянно меняющийся список системных процессов, который сортируется в зависимости от нагрузки на ЦПУ. Если делать сравнение `stop`, то `htop` показывает абсолютно все процессы в системе, время их непрерывного использования, загрузку процессоров и расход оперативной памяти.

- 8) `find` – 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.

Команда `find` имеет такой синтаксис:

`find[папка][параметры] критерий шаблон [действие]`

Папка – каталог в котором будем искать

Параметры – дополнительные параметры, например, глубина поиска, и т.д.

Критерий – по какому критерию будем искать: имя, дата создания, права, владелец и т.д.

Шаблон – непосредственно значение по которому будем отбирать файлы.

Основные параметры:

- P никогда не открывать символические ссылки
- L - получает информацию о файлах по символическим ссылкам. Важно для дальнейшей обработки, чтобы обрабатывалась не ссылка, а сам файл.

- maxdepth - максимальная глубина поиска по подкаталогам, для поиска только в текущем каталоге установите 1.

- depth - искать сначала в текущем каталоге, а потом в подкаталогах

- mount искать файлы только в этой файловой системе.

- version - показать версию утилиты `find`

- print - выводить полные имена файлов

- typef - искать только файлы

- typed - поиск папки в Linux

Основные критерии:

- name - поиск файлов по имени

- perm - поиск файлов в Linux по режиму доступа

- user - поиск файлов по владельцу

- group - поиск по группе

- mtime - поиск по времени модификации файла

- atime - поиск файлов по дате последнего чтения
- nogroup - поиск файлов, не принадлежащих ни одной группе
- nouser - поиск файлов без владельцев
- newer - найти файлы новее чем указанный
- size - поиск файлов в Linux по их размеру

Примеры:

find~ -type d поиск директорий в домашнем каталоге

find~ -type f -name ".*" поиск скрытых файлов в домашнем каталоге

- 9) Файл по его содержимому можно найти с помощью команды grep: «grep -r» слово/выражение, которое нужно найти».
- 10) Утилита df, позволяет проанализировать свободное пространство на всех подключенных к системе разделах.
- 11) При выполнении команды du (без указания папки и опции) можно получить все файлы и папки текущей директории с их размерами. Для домашнего каталога: du ~/
- 12) Основные сигналы (каждый сигнал имеет свой номер), которые используются для завершения процесса:
 - SIGINT–самый безобидный сигнал завершения, означает Interrupt. Он отправляется процессу, запущенному из терминала с помощью сочетания клавиш Ctrl+C. Процесс правильно завершает все свои действия и возвращает управление;
 - SIGQUIT–это еще один сигнал, который отправляется с помощью сочетания клавиш, программе, запущенной в терминале. Он сообщает ей что нужно завершиться и программа может выполнить корректное завершение или проигнорировать сигнал. В отличие от предыдущего, она генерирует дампы памяти. Сочетание клавиш Ctrl+Q;

- SIGHUP–сообщает процессу, что соединение с управляющим терминалом разорвано, отправляется, в основном, системой при разрыве соединения с интернетом;
- SIGTERM–немедленно завершает процесс, но обрабатывается программой, поэтому позволяет ей завершить дочерние процессы и освободить все ресурсы;
- SIGKILL–тоже немедленно завершает процесс, но, в отличие от предыдущего варианта, он не передается самому процессу, а обрабатывается ядром. Поэтому ресурсы и дочерние процессы остаются запущенными.

Также для передачи сигналов процессам в Linux используется утилита `kill`, её синтаксис: `kill [-сигнал] [pid_процесса]` (PID – уникальный идентификатор процесса). Сигнал представляет собой один из выше перечисленных сигналов для завершения процесса.

Перед тем, как выполнить остановку процесса, нужно определить его PID. Для этого используют команды `ps` и `grep`. Команда `ps` предназначена для вывода списка активных процессов в системе и информации о них. Команда `grep` запускается одновременно с `ps` (в канале) и будет выполнять поиск по результатам команды `ps`.

Утилита `pkill` – это оболочка для `kill`, она ведет себя точно так же, и имеет тот же синтаксис, только в качестве идентификатора процесса ей нужно передать его имя.

`killall` работает аналогично двум предыдущим утилитам. Она тоже принимает имя процесса в качестве параметра и ищет его PID в директории `/proc`. Но эта утилита обнаружит все процессы с таким именем и завершит их.