

Липецкий государственный технический университет

Факультет автоматизации и информатики

Кафедра автоматизированных систем управления

Лабораторная работа №3

по Дисциплине «Операционная система Linux»

Планирование процессов

Студент

Глебов Д.А.

Группа АИ-18

Руководитель

Кургасов В.В.

к.п.н.

Липецк 2020 г.

Оглавление	
Задание кафедры.....	3
Ход работы.....	4
Вывод.....	23

Задание кафедры

1. Повторить команды `cat`, `head`, `tail`, `more`, `less`, `grep`, `find`.
2. Разобраться с понятиями конвейер, перенаправление ввода-вывода.
3. Ознакомиться с информацией из рекомендованных источников и других про конвейеризации.
4. Повторить назначение прав доступа. Команды `chmod`, `chown`.
5. Ознакомиться с информацией по теме процессы, посмотреть и опробовать примеры наиболее распространенных команд, изучить возможность запуска процессов в `supervisor`.
6. Изучить возможность автоматического запуска программ по расписанию.

Ход работы

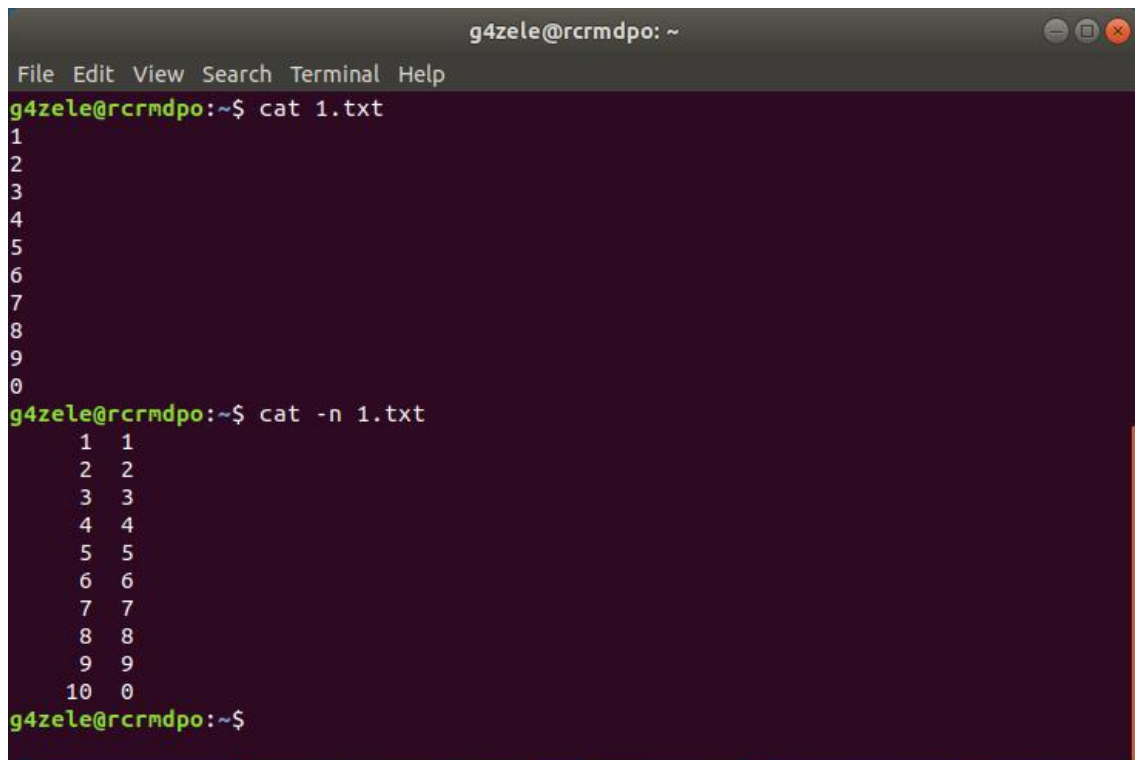
Для начала разберёмся с командой `cat`.

Задача команды `cat` очень проста – она читает данные из файла или стандартного ввода и выводит их на экран. Это все, чем занимается утилита. Но с помощью ее опций и операторов перенаправления вывода можно сделать очень многое. Сначала рассмотрим синтаксис утилиты:

\$ cat опции файл1 файл2 ...

Вы можете передать утилите несколько файлов и тогда их содержимое будет выведено поочередно, без разделителей. Опции позволяют очень сильно видоизменить вывод и сделать именно то, что вам нужно. Рассмотрим основные опции:

- `-b` - нумеровать только непустые строки;
- `-E` - показывать символ \$ в конце каждой строки;
- `-n` - нумеровать все строки;
- `-s` - удалять пустые повторяющиеся строки;
- `-T` - отображать табуляции в виде `^I`;
- `-h` - отобразить справку;
- `-v` - версия утилиты. Пример использования утилиты `cat`:

A terminal window titled 'g4zele@rcrmdpo: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The prompt is 'g4zele@rcrmdpo:~\$'. The command 'cat 1.txt' is entered, and the output is a list of numbers 1 through 10. Then, the command 'cat -n 1.txt' is entered, and the output is a list of numbers 1 through 10, each followed by a space and the same number. The prompt is 'g4zele@rcrmdpo:~\$' again.

```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
g4zele@rcrmdpo:~$ cat 1.txt
1
2
3
4
5
6
7
8
9
0
g4zele@rcrmdpo:~$ cat -n 1.txt
 1 1
 2 2
 3 3
 4 4
 5 5
 6 6
 7 7
 8 8
 9 9
10 0
g4zele@rcrmdpo:~$
```

Рисунок 1 – Результат использования команды cat

Далее перейдём к команде head.

Команда head выводит начальные строки (по умолчанию — 10) из одного или нескольких документов. Также она может показывать данные, которые передает на вывод другая утилита.

Синтаксис у команды head следующий:

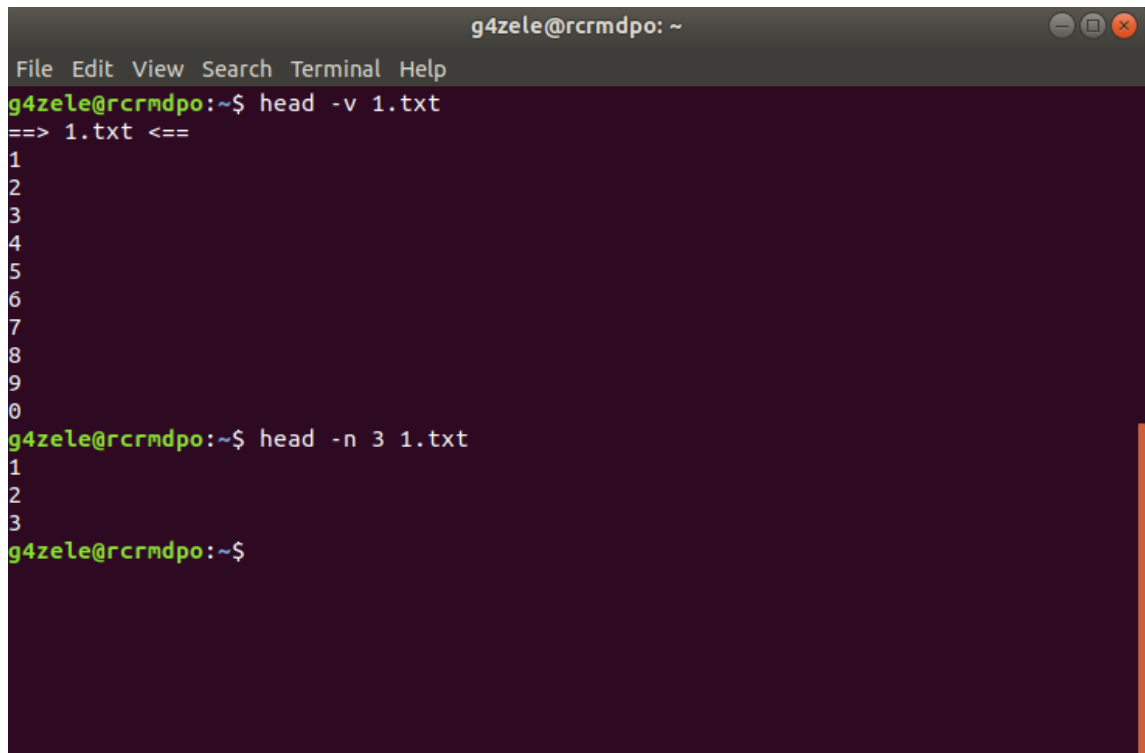
\$ head опции файл

Здесь:

- Опции — это параметр, который позволяет настраивать работу команды таким образом, чтобы результат соответствовал конкретным потребностям пользователя.
- Файл — это имя документа (или имена документов, если их несколько). Если это значение не задано либо вместо него стоит знак «-», команда будет брать данные из стандартного вывода. Чаще всего к команде head применяются такие опции:
 - -c (--bytes) - позволяет задавать количество текста не в строках, а в байтах. При записи в виде --bytes=[-]NUM выводит на экран все содержимое файла, кроме NUM байт, расположенных в конце документа.
 - -n (--lines) — показывает заданное количество строк вместо 10, которые выводятся по умолчанию. Если записать эту опцию в виде --lines=[-]NUM, будет показан весь текст кроме последних NUM строк.
 - -q (--quiet, --silent) — выводит только текст, не добавляя к нему название файла.
 - -v (--verbose) — перед текстом выводит название файла.
 - -z (--zero-terminated) — символы перехода на новую строку заменяет символами завершения строк.

Переменная NUM, упомянутая выше — это любое число от 0 до бесконечности, задаваемое пользователем. Оно может быть обычным либо содержать в себе множитель.

Пример использования команды head:



```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
g4zele@rcrmdpo:~$ head -v 1.txt
==> 1.txt <==
1
2
3
4
5
6
7
8
9
10
g4zele@rcrmdpo:~$ head -n 3 1.txt
1
2
3
g4zele@rcrmdpo:~$
```

Рисунок 2 – Результат выполнения команды head

Команда Tail:

«Tail» позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме.

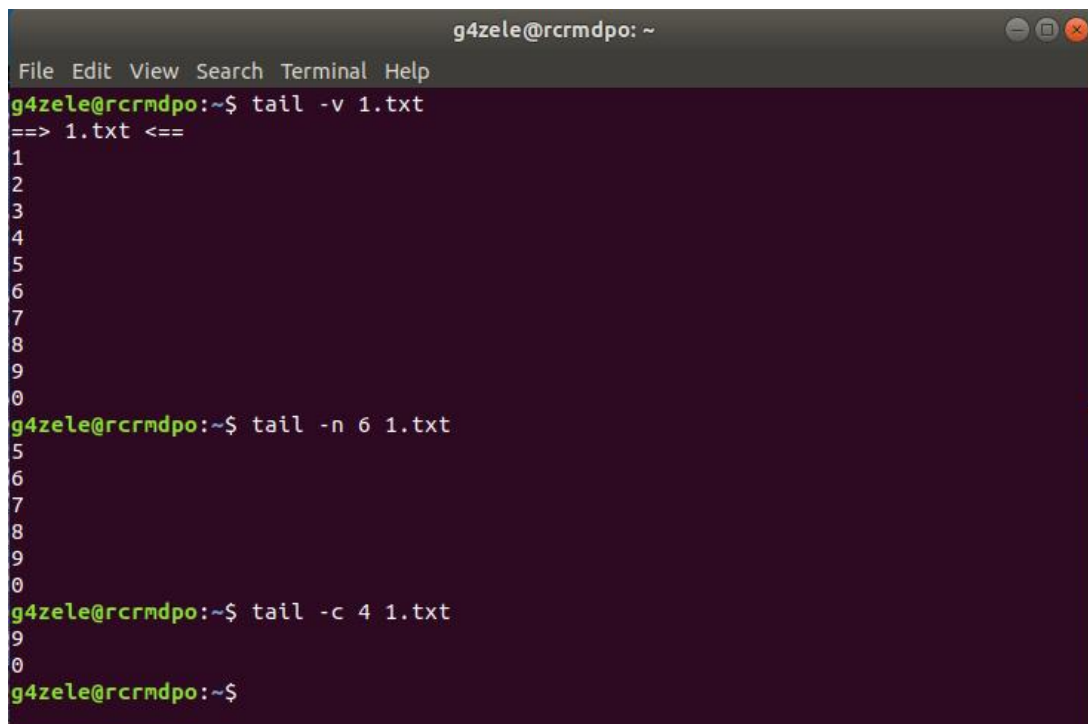
Синтаксис команды очень прост:

\$ tail опции файл

По умолчанию утилита выводит десять последних строк из файла, но ее поведение можно настроить с помощью опций:

- -с - выводить указанное количество байт с конца файла;
- -f - обновлять информацию по мере появления новых строк в файле;
- -n - выводить указанное количество строк из конца файла;
- --pid - используется с опцией -f, позволяет завершить работу утилиты, когда завершится указанный процесс;
- -q - не выводить имена файлов;
- --retry - повторять попытки открыть файл, если он недоступен;
- -v - выводить подробную информацию о файле;

Пример использования команды tail:

A screenshot of a terminal window titled 'g4zele@rcrmdpo: ~'. The terminal shows the execution of the 'tail' command on a file named '1.txt'. The first command is 'tail -v 1.txt', which displays lines 1 through 10 of the file. The second command is 'tail -n 6 1.txt', which displays lines 5 through 10. The third command is 'tail -c 4 1.txt', which displays the last four characters of the file, which are '90'. The prompt 'g4zele@rcrmdpo:~\$' is shown at the end of the third command.

```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
g4zele@rcrmdpo:~$ tail -v 1.txt
==> 1.txt <==
1
2
3
4
5
6
7
8
9
0
g4zele@rcrmdpo:~$ tail -n 6 1.txt
5
6
7
8
9
0
g4zele@rcrmdpo:~$ tail -c 4 1.txt
90
g4zele@rcrmdpo:~$
```

Рисунок 3 – Результат выполнения команды tail

Команде more:

Утилита more предназначена для постраничного просмотра файлов в терминале Linux. Своим названием она обязана надписи more (в русскоязычном варианте – дальше), появляющейся внизу каждой страницы.

Команда more – одна из самых примитивных команд для работы с текстом.

В эмуляторе терминала Linux команда записывается так:

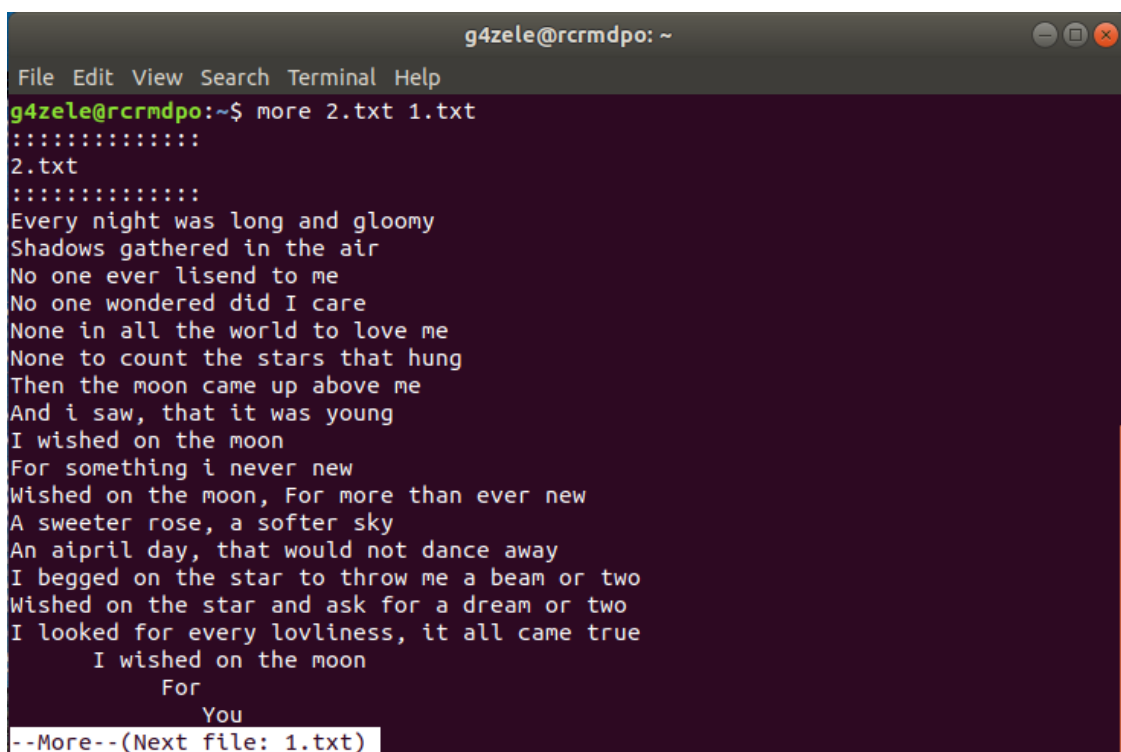
\$ more опции файл

Список опций команды:

- -l — игнорирование в тексте символа разрыва страницы;
- -f — подсчёт числа логических строк вместо экранных;
- -p — очистка экрана терминала для того, чтобы пользователю не пришлось пользоваться прокруткой перед выводом следующей порции текста;
- -c — устранение потребности в прокрутке (как и -p) — отображение текста, начиная с верха экрана, и стирание при этом предыдущего вывода построчно;

- -s — замена нескольких пустых строк, расположенных подряд, одной пустой строкой;
- -u — удаление подчёркивания;
- -n — отображение n-го количества строк;
- +n — отображение текста, начиная со строки с номером n;
- +/строка — поиск в файле указанной строки и начало вывода текста именно с неё;
- --help — вызов справки;
- -v (--version) — вывод на экран текущей версии утилиты.

Пример использования команды more:



```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
g4zele@rcrmdpo:~$ more 2.txt 1.txt
::::::::::::
2.txt
::::::::::::
Every night was long and gloomy
Shadows gathered in the air
No one ever lisend to me
No one wondered did I care
None in all the world to love me
None to count the stars that hung
Then the moon came up above me
And i saw, that it was young
I wished on the moon
For something i never new
Wished on the moon, For more than ever new
A sweeter rose, a softer sky
An ai pril day, that would not dance away
I begged on the star to throw me a beam or two
Wished on the star and ask for a dream or two
I looked for every lovliness, it all came true
    I wished on the moon
        For
            You
--More--(Next file: 1.txt)
```

Рисунок 4 – Результат открытия двух файлов командой more

Похожей на more является команда less.

Особенность less заключается в том, что команда не считывает текст полностью, а загружает его небольшими фрагментами.

Запись команды less в терминале выглядит так:

less опции файл

Наиболее популярные опции:

- -a, --search-skip-screen — не осуществлять поиск в тексте, который в данный момент отображен на экране;
- -bn, --buffers=n — задать размер буфера памяти;
- -c, --clear-screen — листать текст, полностью стирая содержимое экрана (построчная прокрутка работать не будет);
- -Dxcolor, --color=xcolor — задать цвет отображаемого текста;
- -E, --QUIT-AT-EOF — выйти, когда утилита достигнет конца файла;
- -e, --quit-at-eof — выйти, когда утилита второй раз достигнет конца файла;
- -F, --quit-if-one-screen — выйти, если содержимое файла помещается на одном экране;
- -f, --force — открыть специальный файл;
- -hn, --max-back-scroll=n — задать максимальное количество строк для прокрутки назад;
- -yn, --max-forw-scroll=n — задать максимальное количество строк для прокрутки вперёд;
- -i, --ignore-case — игнорировать регистр;
- -I, --IGNORE-CASE — игнорировать регистр, даже если паттерн для поиска содержит заглавные буквы;
- -jn, --jump-target=n — указать, в какой строке должна быть выведена искомая информация;
- -J, --status-column — пометить строки, соответствующие результатам поиска;

- -n, --line-numbers — не выводить номера строк;
- -N, --LINE-NUMBERS — вывести номера строк;
- -s, --squeeze-blank-lines — заменить множество идущих подряд пустых строк одной пустой строкой;
- -w, --hilite-unread — выделить первую строку нового фрагмента текста.

Пример использования команды less:

```

g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
 1 Every night was long and gloomy
 2 Shadows gathered in the air
 3 No one ever lisend to me
 4 No one wondered did I care
 5 None in all the world to love me
 6 None to count the stars that hung
 7 Then the moon came up above me
 8 And i saw, that it was young
 9 I wished on the moon
10 For something i never new
11 Wished on the moon, For more than ever new
12 A sweeter rose, a softer sky
13 An ai pril day, that would not dance away
14 I begged on the star to throw me a beam or two
15 Wished on the star and ask for a dream or two
16 I looked for every lovliness, it all came true
17 I wished on the moon
18 For
19 You
~
~
~
(END)

```

Рисунок 5 – Результат открытия файла командой less -N 2.txt

Команда Grep:

Утилита grep решает множество задач, в основном она используется для поиска строк, соответствующих строке в тексте или содержимому файлов. Также она может находить по шаблону или регулярным выражениям. Команда в считанные секунды найдёт файл с нужной строчкой, текст в файле или отфильтрует из вывода только пару нужных строк.

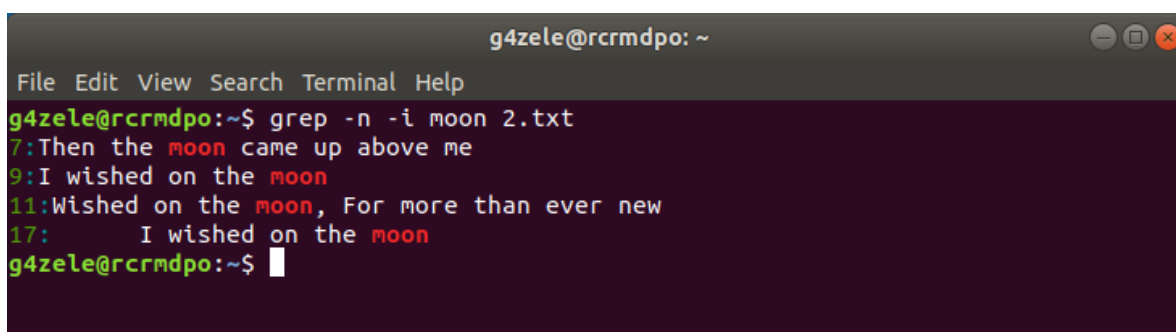
Синтаксис команды выглядит следующим образом:

\$ grep [опции] шаблон [имя файла...]

Давайте рассмотрим самые основные опции утилиты, которые помогут более эффективно выполнять поиск текста в файлах `grep`:

- `-b` - показывать номер блока перед строкой;
- `-c` - подсчитать количество вхождений шаблона;
- `-h` - не выводить имя файла в результатах поиска внутри файлов Linux;
- `-i` - не учитывать регистр;
- `-l` - отобразить только имена файлов, в которых найден шаблон;
- `-n` - показывать номер строки в файле;
- `-s` - не показывать сообщения об ошибках;
- `-v` - инвертировать поиск, выдавать все строки кроме тех, что содержат шаблон;
- `-w` - искать шаблон как слово, окружённое пробелами;
- `-e` - использовать регулярные выражения при поиске;
- `-An` - показать вхождение и `n` строк до него;
- `-Bn` - показать вхождение и `n` строк после него;
- `-Cn` - показать `n` строк до и после вхождения;

Пример использования команды `grep`:



```
g4zele@rcrmdpo: ~  
File Edit View Search Terminal Help  
g4zele@rcrmdpo:~$ grep -n -i moon 2.txt  
7:Then the moon came up above me  
9:I wished on the moon  
11:Wished on the moon, For more than ever new  
17:    I wished on the moon  
g4zele@rcrmdpo:~$
```

Рисунок 6 – Результат поиска слова `moon` в файле

Find - это одна из наиболее важных и часто используемых утилит системы Linux.

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.

Утилита `find` предустановлена по умолчанию во всех Linux дистрибутивах, поэтому вам не нужно будет устанавливать никаких дополнительных пакетов. Это очень важная находка для тех, кто хочет использовать командную строку наиболее эффективно.

Команда `find` имеет такой синтаксис:

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

Рассмотрим самые полезные 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

- `-type f` - искать только файлы

- `-type d` - поиск папки в Linux Критериев у команды `find` в Linux очень много, рассмотрим основные:

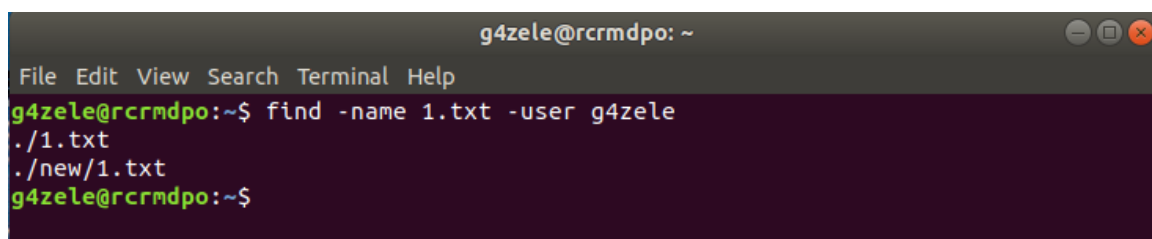
- `-name` - поиск файлов по имени

- `-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

Пример использования команды find:

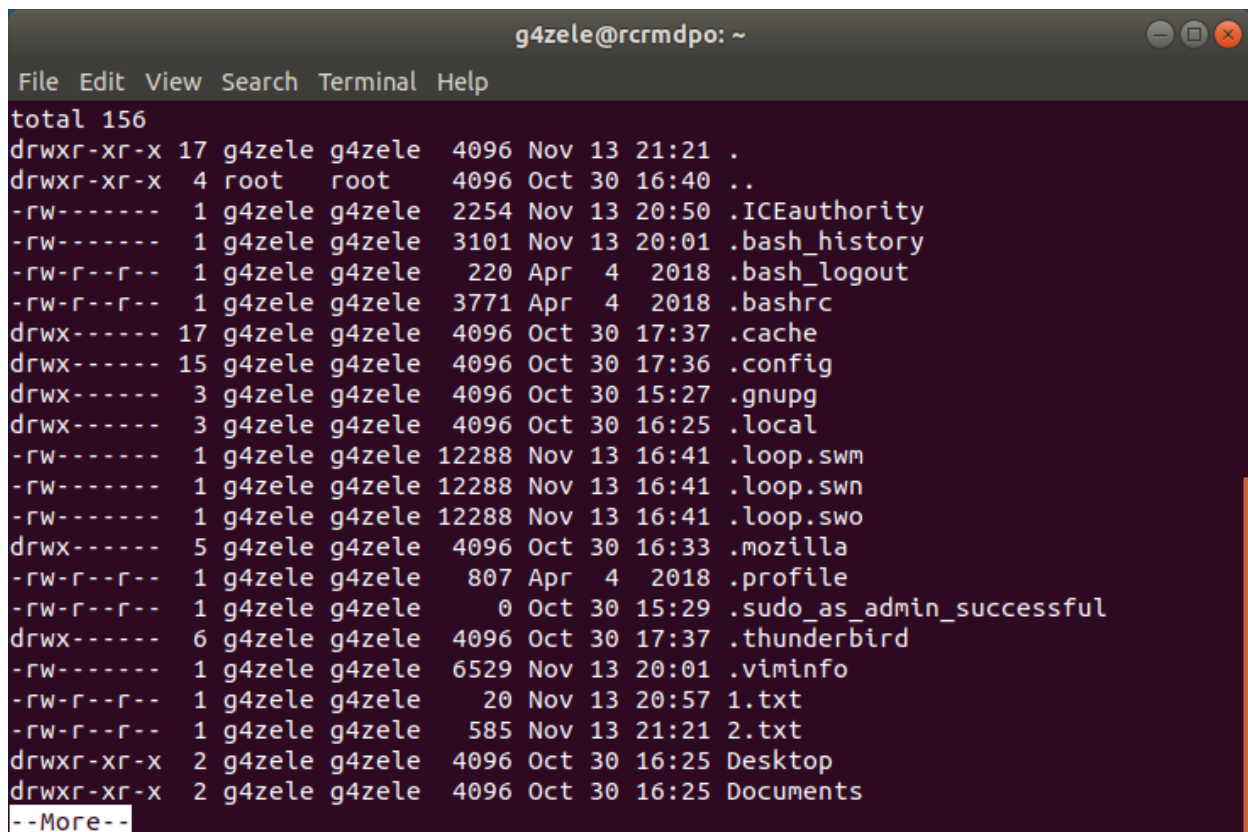
A screenshot of a terminal window titled 'g4zele@rcrmdpo: ~'. The terminal has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The prompt is 'g4zele@rcrmdpo:~\$'. The command 'find -name 1.txt -user g4zele' has been entered. The output shows two lines: './1.txt' and './new/1.txt'. The prompt 'g4zele@rcrmdpo:~\$' is shown again at the bottom.

```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
g4zele@rcrmdpo:~$ find -name 1.txt -user g4zele
./1.txt
./new/1.txt
g4zele@rcrmdpo:~$
```

Рисунок 7 – Результат использования команды find для поиска файла

Теперь разберёмся с понятием конвейера. Дело в том, что Linux поддерживает использование состыкованных команд, что обычно называется конвейером. Стыковку осуществляет командная оболочка, которая stdout первой команды направляет на stdin второй команды. Для стыковки используется символ |.

Например, мы можем организовать постраничный просмотр какойнибудь директории большого размера. Сделаем это с помощью состыкованных команд. Напишем в терминале `ls -la | more` и посмотрим результат:



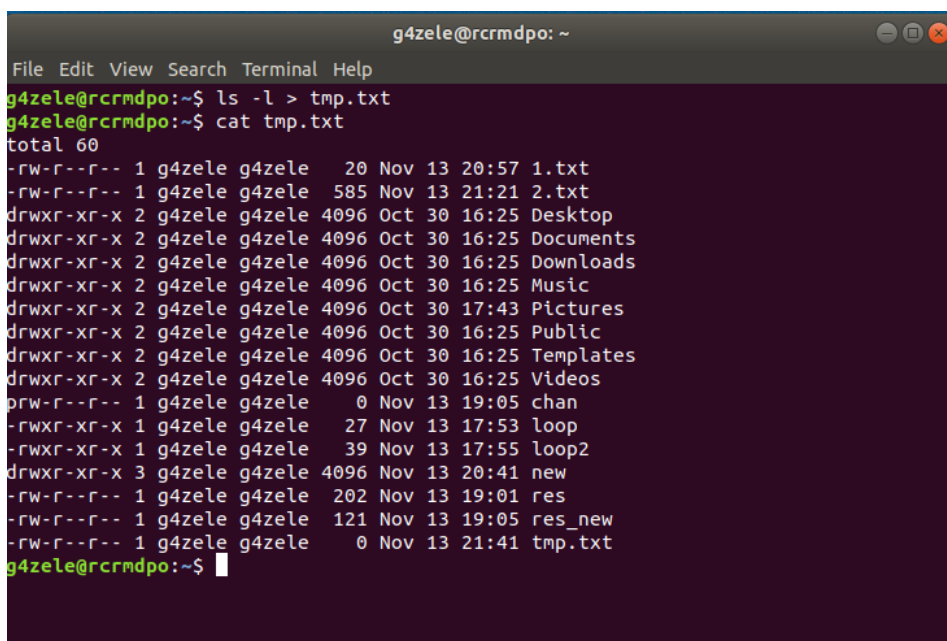
```
g4zele@rcrmdpo: ~
File Edit View Search Terminal Help
total 156
drwxr-xr-x 17 g4zele g4zele 4096 Nov 13 21:21 .
drwxr-xr-x  4 root   root   4096 Oct 30 16:40 ..
-rw-r--r--  1 g4zele g4zele 2254 Nov 13 20:50 .ICEauthority
-rw-r--r--  1 g4zele g4zele 3101 Nov 13 20:01 .bash_history
-rw-r--r--  1 g4zele g4zele  220 Apr  4 2018 .bash_logout
-rw-r--r--  1 g4zele g4zele 3771 Apr  4 2018 .bashrc
drwxr-xr-x 17 g4zele g4zele 4096 Oct 30 17:37 .cache
drwxr-xr-x 15 g4zele g4zele 4096 Oct 30 17:36 .config
drwxr-xr-x  3 g4zele g4zele 4096 Oct 30 15:27 .gnupg
drwxr-xr-x  3 g4zele g4zele 4096 Oct 30 16:25 .local
-rw-r--r--  1 g4zele g4zele 12288 Nov 13 16:41 .loop.swm
-rw-r--r--  1 g4zele g4zele 12288 Nov 13 16:41 .loop.swn
-rw-r--r--  1 g4zele g4zele 12288 Nov 13 16:41 .loop.swo
drwxr-xr-x  5 g4zele g4zele 4096 Oct 30 16:33 .mozilla
-rw-r--r--  1 g4zele g4zele  807 Apr  4 2018 .profile
-rw-r--r--  1 g4zele g4zele    0 Oct 30 15:29 .sudo_as_admin_successful
drwxr-xr-x  6 g4zele g4zele 4096 Oct 30 17:37 .thunderbird
-rw-r--r--  1 g4zele g4zele 6529 Nov 13 20:01 .viminfo
-rw-r--r--  1 g4zele g4zele  20 Nov 13 20:57 1.txt
-rw-r--r--  1 g4zele g4zele  585 Nov 13 21:21 2.txt
drwxr-xr-x  2 g4zele g4zele 4096 Oct 30 16:25 Desktop
drwxr-xr-x  2 g4zele g4zele 4096 Oct 30 16:25 Documents
--More--
```

Рисунок 8 – Использование конвейера

Затем нам следует разобраться с перенаправлением ввода-вывода в ОС Linux.

Если говорить о перенаправлении вывода, то всё очень просто. Вы можете перенаправить вывод в файл с помощью символа >.

Например, перенаправим вывод содержимого директории в файл:

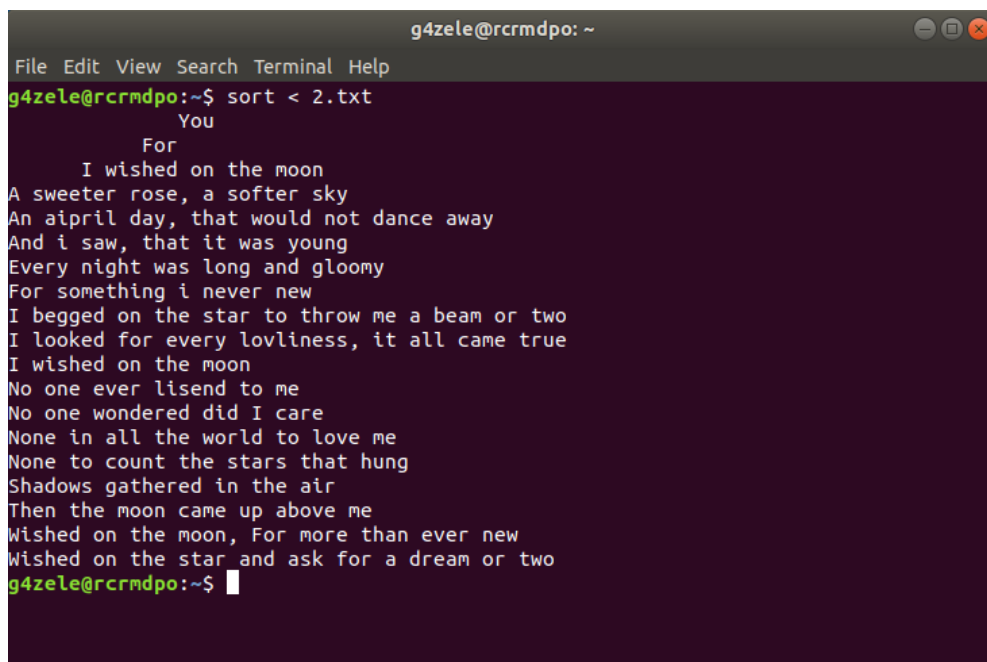
A terminal window titled 'g4zele@rcrmdpo: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The user enters 'ls -l > tmp.txt' and then 'cat tmp.txt'. The output shows a detailed directory listing for the home directory, including files like 1.txt, 2.txt, Desktop, Documents, Downloads, Music, Pictures, Public, Templates, Videos, and various folders like chan, loop, loop2, new, res, res_new, and tmp.txt.

```
g4zele@rcrmdpo:~$ ls -l > tmp.txt
g4zele@rcrmdpo:~$ cat tmp.txt
total 60
-rw-r--r-- 1 g4zele g4zele  20 Nov 13 20:57 1.txt
-rw-r--r-- 1 g4zele g4zele 585 Nov 13 21:21 2.txt
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Desktop
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Documents
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Downloads
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Music
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 17:43 Pictures
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Public
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Templates
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Videos
prw-r--r-- 1 g4zele g4zele   0 Nov 13 19:05 chan
-rwxr-xr-x 1 g4zele g4zele  27 Nov 13 17:53 loop
-rwxr-xr-x 1 g4zele g4zele  39 Nov 13 17:55 loop2
drwxr-xr-x 3 g4zele g4zele 4096 Nov 13 20:41 new
-rw-r--r-- 1 g4zele g4zele 202 Nov 13 19:01 res
-rw-r--r-- 1 g4zele g4zele 121 Nov 13 19:05 res_new
-rw-r--r-- 1 g4zele g4zele   0 Nov 13 21:41 tmp.txt
g4zele@rcrmdpo:~$
```

Рисунок 9 – Перенаправление вывода

Перенаправление ввода же реализуется с помощью символа <.

Например, нам нужно отсортировать содержимое файла. Сделаем это с помощью перенаправления ввода:

A terminal window titled 'g4zele@rcrmdpo: ~' with a menu bar (File, Edit, View, Search, Terminal, Help). The user enters 'sort < 2.txt'. The output shows the contents of 2.txt sorted alphabetically, which is the lyrics of the poem 'The Moon Song' by John Keats.

```
g4zele@rcrmdpo:~$ sort < 2.txt
You
For
I wished on the moon
A sweeter rose, a softer sky
An ai pril day, that would not dance away
And i saw, that it was young
Every night was long and gloomy
For something i never new
I begged on the star to throw me a beam or two
I looked for every lovliness, it all came true
I wished on the moon
No one ever lisend to me
No one wondered did I care
None in all the world to love me
None to count the stars that hung
Shadows gathered in the air
Then the moon came up above me
Wished on the moon, For more than ever new
Wished on the star and ask for a dream or two
g4zele@rcrmdpo:~$
```

Рисунок 10 – Перенаправление ввода

Теперь поговорим о командах `chmod` и `chown` (мы уже говорили о них первой лабораторной работе).

Начнём с команды `chmod`.

Эта команда имеет типичный для команд `linux` синтаксис, сначала команда, затем опции, а в конце файл или папка, к которой ее нужно применить:

\$ chmod опции права /путь/к/файлу

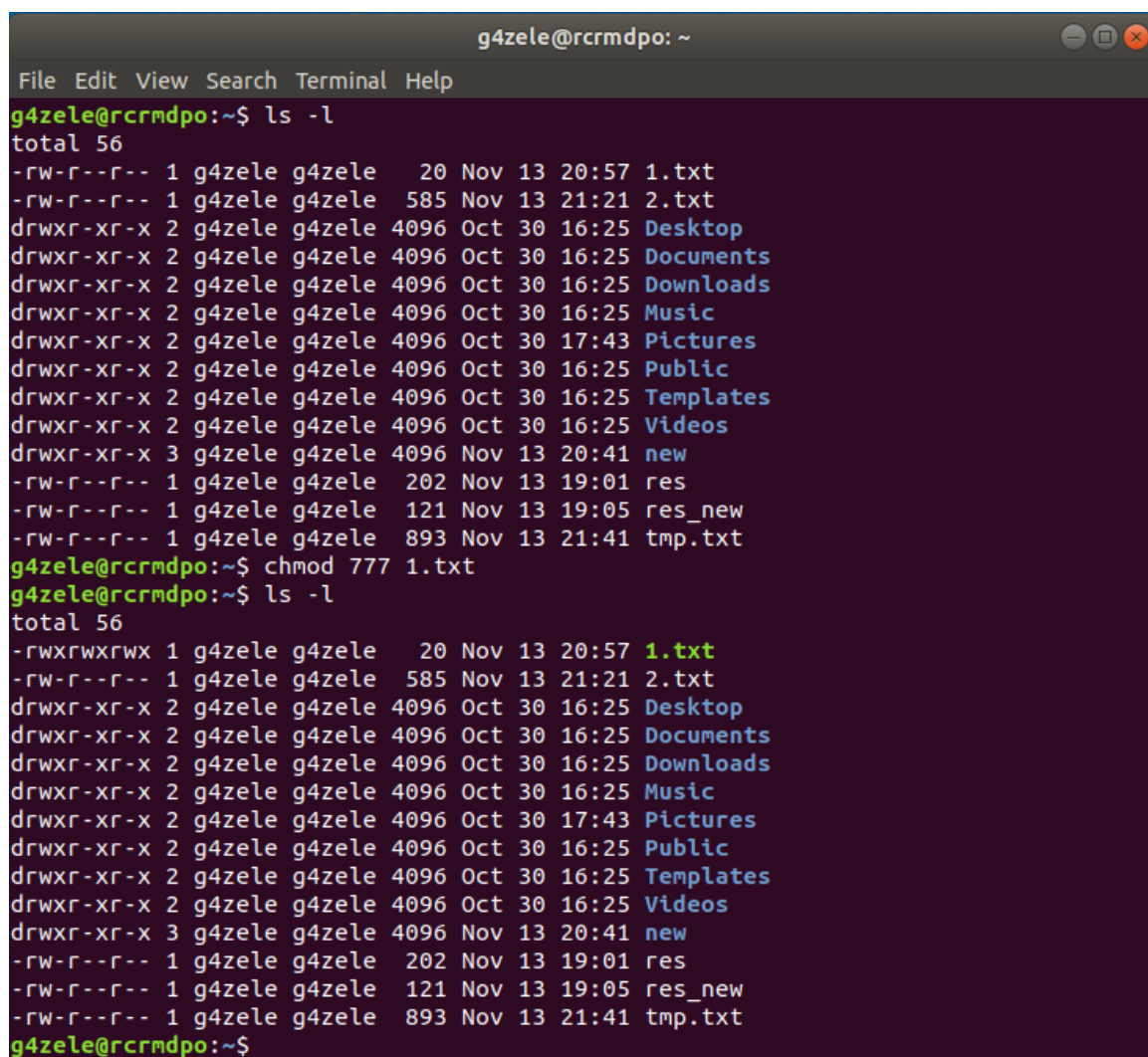
Сначала рассмотрим какими бывают права доступа `linux` и как они устанавливаются. Пред этим рекомендую прочитать статью про права, ссылка на которую есть выше. Есть три основных вида прав:

- `r` - чтение;
- `w` - запись;
- `x` - выполнение;
- `s` - выполнение от имени суперпользователя (дополнительный);

Права можно назначать следующим образом: для начала нам нужно определить, для какой категории пользователя (владелец файла, группа файла, остальные пользователи) какие нужно установить права, затем установим права с помощью восьмеричного шаблона:

- `0` - никаких прав;
- `1` - только выполнение;
- `2` - только запись;
- `3` - выполнение и запись;
- `4` - только чтение;
- `5` - чтение и выполнение;
- `6` - чтение и запись;
- `7` - чтение, запись и выполнение.

Приведём пример выполнения команды `chmod`:



```
g4zele@rcrmdpo: ~  
File Edit View Search Terminal Help  
g4zele@rcrmdpo:~$ ls -l  
total 56  
-rw-r--r-- 1 g4zele g4zele  20 Nov 13 20:57 1.txt  
-rw-r--r-- 1 g4zele g4zele 585 Nov 13 21:21 2.txt  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Desktop  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Documents  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Downloads  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Music  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 17:43 Pictures  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Public  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Templates  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Videos  
drwxr-xr-x 3 g4zele g4zele 4096 Nov 13 20:41 new  
-rw-r--r-- 1 g4zele g4zele  202 Nov 13 19:01 res  
-rw-r--r-- 1 g4zele g4zele  121 Nov 13 19:05 res_new  
-rw-r--r-- 1 g4zele g4zele  893 Nov 13 21:41 tmp.txt  
g4zele@rcrmdpo:~$ chmod 777 1.txt  
g4zele@rcrmdpo:~$ ls -l  
total 56  
-rwxrwxrwx 1 g4zele g4zele  20 Nov 13 20:57 1.txt  
-rw-r--r-- 1 g4zele g4zele 585 Nov 13 21:21 2.txt  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Desktop  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Documents  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Downloads  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Music  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 17:43 Pictures  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Public  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Templates  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Videos  
drwxr-xr-x 3 g4zele g4zele 4096 Nov 13 20:41 new  
-rw-r--r-- 1 g4zele g4zele  202 Nov 13 19:01 res  
-rw-r--r-- 1 g4zele g4zele  121 Nov 13 19:05 res_new  
-rw-r--r-- 1 g4zele g4zele  893 Nov 13 21:41 tmp.txt  
g4zele@rcrmdpo:~$
```

Рисунок 11 – Изменение прав на файл

Здесь мы установили права на чтение, запись и исполнение файла для всех категорий пользователя: первая семёрка открывает доступ для создателя файла, вторая – для группы файла, третья – для остальных пользователей.

Команда `chown`.

Синтаксис `chown` прост:

\$ `chown` пользователь опции /путь/к/файлу

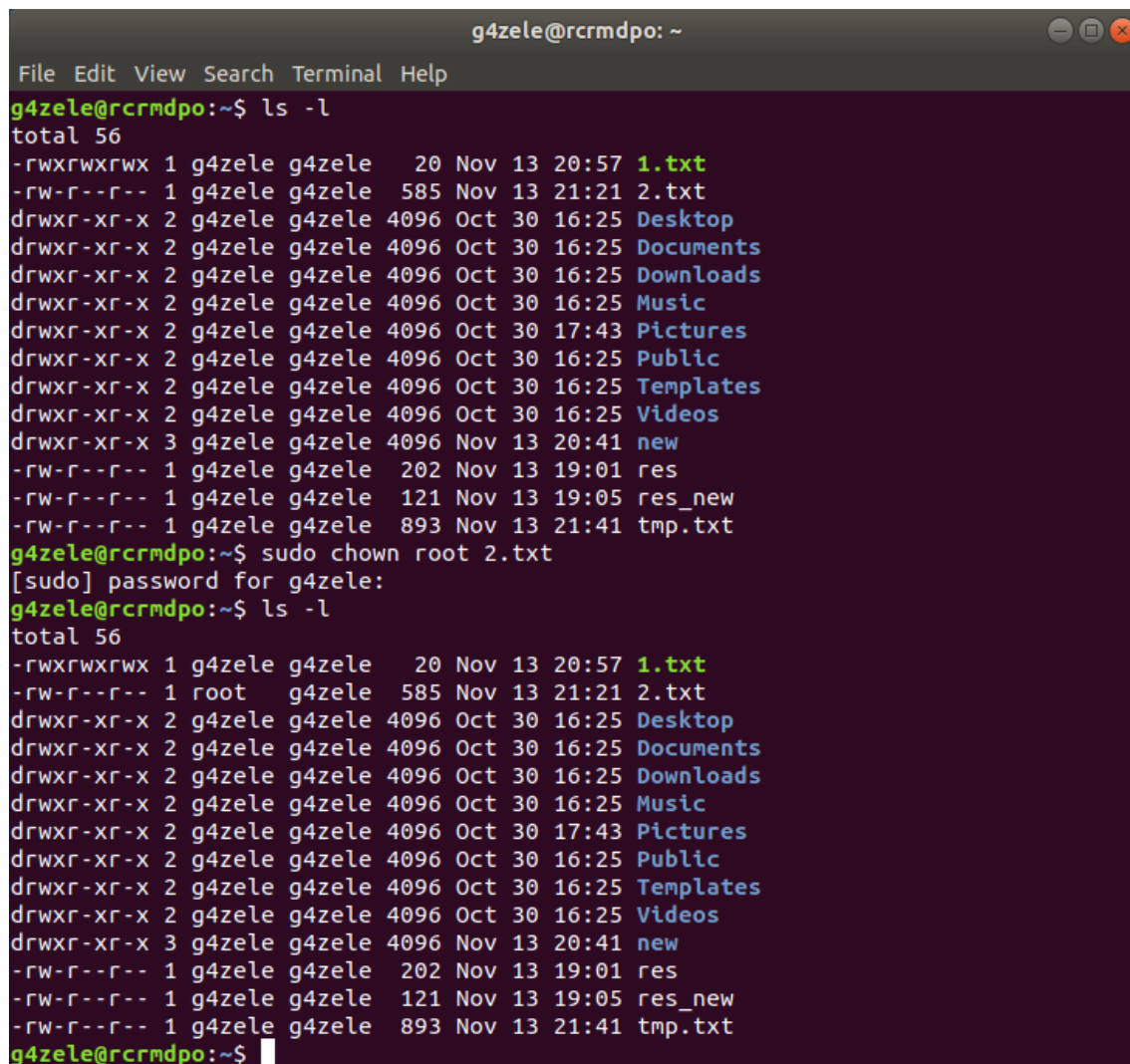
В поле пользователь надо указать пользователя, которому мы хотим передать файл.

Вот основные опции, которые могут вам понадобиться:

- `-c, --changes` - подробный вывод всех выполняемых изменений;

- -f, --silent, --quiet - минимум информации, скрыть сообщения об ошибках;
- --dereference - изменять права для файла к которому ведет символическая ссылка вместо самой ссылки (поведение по умолчанию);
- -h, --no-dereference - изменять права символических ссылок и не трогать файлы, к которым они ведут;
- --from - изменять пользователя только для тех файлов, владельцем которых является указанный пользователь и группа;
- -R, --recursive - рекурсивная обработка всех подкаталогов;
- -H - если передана символическая ссылка на директорию - перейти по ней;
- -L - переходить по всем символическим ссылкам на директории;
- -P - не переходить по символическим ссылкам на директории (по умолчанию).

Изменим владельца файла:



```
g4zele@rcrmdpo: ~  
File Edit View Search Terminal Help  
g4zele@rcrmdpo:~$ ls -l  
total 56  
-rwxrwxrwx 1 g4zele g4zele  20 Nov 13 20:57 1.txt  
-rw-r--r-- 1 g4zele g4zele 585 Nov 13 21:21 2.txt  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Desktop  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Documents  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Downloads  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Music  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 17:43 Pictures  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Public  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Templates  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Videos  
drwxr-xr-x 3 g4zele g4zele 4096 Nov 13 20:41 new  
-rw-r--r-- 1 g4zele g4zele  202 Nov 13 19:01 res  
-rw-r--r-- 1 g4zele g4zele  121 Nov 13 19:05 res_new  
-rw-r--r-- 1 g4zele g4zele  893 Nov 13 21:41 tmp.txt  
g4zele@rcrmdpo:~$ sudo chown root 2.txt  
[sudo] password for g4zele:  
g4zele@rcrmdpo:~$ ls -l  
total 56  
-rwxrwxrwx 1 g4zele g4zele  20 Nov 13 20:57 1.txt  
-rw-r--r-- 1 root   g4zele 585 Nov 13 21:21 2.txt  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Desktop  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Documents  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Downloads  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Music  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 17:43 Pictures  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Public  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Templates  
drwxr-xr-x 2 g4zele g4zele 4096 Oct 30 16:25 Videos  
drwxr-xr-x 3 g4zele g4zele 4096 Nov 13 20:41 new  
-rw-r--r-- 1 g4zele g4zele  202 Nov 13 19:01 res  
-rw-r--r-- 1 g4zele g4zele  121 Nov 13 19:05 res_new  
-rw-r--r-- 1 g4zele g4zele  893 Nov 13 21:41 tmp.txt  
g4zele@rcrmdpo:~$
```

Рисунок 12 – Использование chown

Теперь перейдём к рассмотрению наиболее распространённых команд для процессов и поговорим о команде top.

Она, в отличие от команды ps, позволяет выводить информацию о системе, а также список процессов динамически обновляя информацию о потребляемых ими ресурсах.

Поговорим про интерактивные команды, которые вы можете выполнять во время работы программы.

- ♣ h - вывод справки по утилите;
- ♣ q или Esc - выход из top;
- ♣ A - выбор цветовой схемы;
- ♣ d или s - изменить интервал обновления информации;

- ♣ Н - выводить потоки процессов;
- ♣ k - послать сигнал завершения процессу;
- ♣ W - записать текущие настройки программы в конфигурационный файл;
- ♣ Y - посмотреть дополнительные сведения о процессе, открытые файлы, порты, логи и т д;
- ♣ Z - изменить цветовую схему;
- ♣ l - скрыть или вывести информацию о средней нагрузке на систему;
- ♣ m - выключить или переключить режим отображения информации о памяти;
- ♣ x - выделять жирным колонку, по которой выполняется сортировка;
- ♣ y - выделять жирным процессы, которые выполняются в данный момент;
- ♣ z - переключение между цветным и одноцветным режимами;
- ♣ c - переключение режима вывода команды, доступен полный путь и только команда;
- ♣ F - настройка полей с информацией о процессах;
- ♣ o - фильтрация процессов по произвольному условию;
- ♣ u - фильтрация процессов по имени пользователя;
- ♣ V - отображение процессов в виде дерева;
- ♣ i - переключение режима отображения процессов, которые сейчас не используют ресурсы процессора;
- ♣ n - максимальное количество процессов, для отображения в программе;
- ♣ L - поиск по слову;
- ♣ < > - перемещение поля сортировки вправо и влево;

Теперь напишем о возможности запуска процессов в supervisor.

Supervisor – это менеджер процессов, который существенно упрощает управление долго работающими программами, предоставляя простой и

понятный интерфейс. Данная утилита упрощает работу с процессами и позволяет администратору контролировать процессы. Помимо этого, он способен обеспечить бесперебойную работу веб-сервиса.

Если говорить о возможности запуска процессов по расписанию, то для этого используется утилита cron.

Cron – программа, предназначенная для выполнения заданий в определенное время, или через определенные промежутки времени.

Для настройки времени, даты и интервала, когда нужно выполнять задание, используется команда crontab, которую рекомендуется выполнять с опцией -e.

Давайте рассмотрим синтаксис настройки одной задачи cron:

минута час день месяц день_недели /путь/к/исполняемому/файлу

Нужно сказать, что обязательно нужно писать полный путь к команде, потому что для команд, запускаемых от имени cron переменная среды PATH будет отличаться, и сервис просто не сможет найти вашу команду. Это вторая самая распространенная причина проблем с Cron. Дата и время указываются с помощью цифр или символа '*'. Этот символ означает, что нужно выполнять каждый раз, если в первом поле - то каждую минуту и так далее.

Данная программа является ультимативным решением для планирования автоматических задач.

Вывод

В ходе выполнения лабораторной работы были закреплены знания по командам `cat`, `head`, `tail`, `more`, `less`, `grep`, `find`, `chown`, `chmod`. Также изучено понятие конвейера и основные принципы перенаправления ввода-вывода. Изучены утилиты `top` для управления процессами, `cron` для планирования процессов и `supervisor` для администрирования.