

Отчёт по лабораторной работе №6

Дисциплина: Операционные системы

Студент: Джеффри Родригес Сантос

Группа: НПМбд-02-20

МОСКВА 2021 г.

1. Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

2. Задание

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
 1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.
 2. В домашнем каталоге создайте директорию `~/ski.places`.
 3. Переместите файл `equipment` в каталог `~/ski.places`.
 4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:
 4. `drwxr-r- ... australia`
 5. `drwx-x-x ... play`

6. `-r-xr-r- ... my_os`

7. `-rw-rw-r- ... feathers`

При необходимости создайте нужные файлы.

8. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

1. Просмотрите содержимое файла `/etc/password`.

2. Скопируйте файл `~/feathers` в файл `~/file.old`.

3. Переместите файл `~/file.old` в каталог `~/play`.

4. Скопируйте каталог `~/play` в каталог `~/fun`.

5. Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.

6. Лишите владельца файла `~/feathers` права на чтение.

7. Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`?

8. Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`?

9. Дайте владельцу файла `~/feathers` право на чтение.

10. Лишите владельца каталога `~/play` права на выполнение.

11. Перейдите в каталог `~/play`. Что произошло?

12. Дайте владельцу каталога `~/play` право на выполнение.

9. Прочитайте ман по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

3. Выполнение лабораторной работы

1. Для начала выполним примеры, описанные в первой части описания лабораторной работы (рис. -fig. 3.1):

1. Скопируем файл `~/abc1` в файл `april` и в файл `may`. Для этого создадим файл `abc1`, используя команду `touch abc1`, далее осуществим копирование с помощью команд `cp abc1 april` и `cp abc1 may`.
2. Скопируем файлы `april` и `may` в каталог `monthly`, используя команды `mkdir monthly` – для создания каталога `monthly` и `cp april may monthly` – для копирования.
3. Скопируем файл `monthly/may` в файл с именем `june`. Выполним команды `cp monthly/may monthly/june` и `ls monthly` (для просмотра содержимого каталога).
4. Скопируем каталог `monthly` в каталог `monthly.00`. Для этого создадим каталог `monthly.00` командой `mkdir monthly.00` и осуществим копирование, используя команду `cp -r monthly monthly.00` (команда `cp` с опцией `r` (`recursive`) позволяет копировать каталоги вместе с входящими в них файлами и каталогами).
5. Скопируем каталог `monthly.00` в каталог `/tmp`, используя команду `cp -r monthly.00 /tmp`.

```

jeffrey@jeffrey-VirtualBox:~$ touch abc1
jeffrey@jeffrey-VirtualBox:~$ cp abc1 april
jeffrey@jeffrey-VirtualBox:~$ cp abc1 may
jeffrey@jeffrey-VirtualBox:~$ ls
abc1    Documents  initial-commit  may    Pictures  Templates
april   Downloads  laboratory      Music  Public    Videos
Desktop helloworld laboratory2     newdir snap
jeffrey@jeffrey-VirtualBox:~$ ls april
april
jeffrey@jeffrey-VirtualBox:~$ cd april; ls; cd ~/may; ls
bash: cd: april: Not a directory
abc1    Documents  initial-commit  may    Pictures  Templates
april   Downloads  laboratory      Music  Public    Videos
Desktop helloworld laboratory2     newdir snap
bash: cd: /home/jeffrey/may: Not a directory
abc1    Documents  initial-commit  may    Pictures  Templates
april   Downloads  laboratory      Music  Public    Videos
Desktop helloworld laboratory2     newdir snap
jeffrey@jeffrey-VirtualBox:~$ mkdir monthly
jeffrey@jeffrey-VirtualBox:~$ cp april may monthly
jeffrey@jeffrey-VirtualBox:~$ ls
abc1    Documents  initial-commit  may    newdir    snap
april   Downloads  laboratory      monthly Pictures  Templates
Desktop helloworld laboratory2     newdir    snap
jeffrey@jeffrey-VirtualBox:~$

```

```

jeffrey@jeffrey-VirtualBox:~$ ls monthly
april may
jeffrey@jeffrey-VirtualBox:~$ cp monthly/may monthly/june
jeffrey@jeffrey-VirtualBox:~$ ls monthly
april june may
jeffrey@jeffrey-VirtualBox:~$ mkdir monthly.00
jeffrey@jeffrey-VirtualBox:~$ cp -r monthly monthly.00
jeffrey@jeffrey-VirtualBox:~$ ls
abc1    Documents  initial-commit  may    Music    Public    Videos
april   Downloads  laboratory      monthly newdir    snap
Desktop helloworld laboratory2     monthly.00 Pictures  Templates
jeffrey@jeffrey-VirtualBox:~$ ls monthly
april june may
jeffrey@jeffrey-VirtualBox:~$ ls monthly.00
monthly
jeffrey@jeffrey-VirtualBox:~$ cp -r monthly.00 /tmp
jeffrey@jeffrey-VirtualBox:~$ ls /tmp
config-err-apfxIw
monthly.00
snap.snap-store
ssh-xmprMSQ8mLch
systemd-private-413fd8392eb84b6c899557622b12ba1e-colord.service-3ilmlj
systemd-private-413fd8392eb84b6c899557622b12ba1e-geoclue.service-qRAkvg
systemd-private-413fd8392eb84b6c899557622b12ba1e-ModemManager.service-hq28Xh
systemd-private-413fd8392eb84b6c899557622b12ba1e-switcheroo-control.service-12F

```

Выполним примеры, описанные в лабораторной работе

```

Lxf
systemd-private-413fd8392eb84b6c899557622b12ba1e-systemd-logind.service-JcEQHh
systemd-private-413fd8392eb84b6c899557622b12ba1e-systemd-resolved.service-T72rx
f
systemd-private-413fd8392eb84b6c899557622b12ba1e-systemd-timesyncd.service-xyup
Wh
systemd-private-413fd8392eb84b6c899557622b12ba1e-upower.service-ooll1h
Temp-0853e8a4-f1ce-4c2f-8cb7-36d78e193ea9
Temp-62ce932d-5914-4387-938b-afba977d03ee
tracker-extract-files.1000
VMwareDnD
jeffrey@jeffrey-VirtualBox:~$

```

```

jeffrey@jeffrey-VirtualBox:~$ cd
jeffrey@jeffrey-VirtualBox:~$ mv april july
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      helloworld    laboratory2  Music      snap
Desktop   initial-commit may          newdir     Templates
Documents july          monthly     Pictures   Videos
Downloads laboratory    monthly.00  Public
jeffrey@jeffrey-VirtualBox:~$ mv july monthly.00
jeffrey@jeffrey-VirtualBox:~$ ls monthly.00
july  monthly
jeffrey@jeffrey-VirtualBox:~$ mv monthly.00/july monthly.00/monthly
jeffrey@jeffrey-VirtualBox:~$ ls monthly.00/monthly
april  july  june  may
jeffrey@jeffrey-VirtualBox:~$ mv monthly.00 monthly.01
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads    laboratory  monthly    newdir     snap
Desktop   helloworld   laboratory2 monthly.01  Pictures   Templates
Documents initial-commit may          Music      Public     Videos
jeffrey@jeffrey-VirtualBox:~$ mkdir reports
jeffrey@jeffrey-VirtualBox:~$ mv monthly.01 reports
jeffrey@jeffrey-VirtualBox:~$ ls reports
monthly.01
jeffrey@jeffrey-VirtualBox:~$

```

1. Изменим название файла `april` на `july` в домашнем каталоге, используя команду «`mv april july`».
2. Переместим файл `july` в каталог `monthly.00` с помощью команды «`mv july monthly.00`». Проверим результат командой «`ls monthly.00`».

3. Переименуем каталог `monthly.00` в `monthly.01`, используя команду «`mv monthly.00 monthly.01`».
4. Переместим каталог `monthly.01` в каталог `reports`. Для этого создадим каталог `reports` с помощью команды «`mkdir reports`» и выполним перемещение командой «`mv monthly.01 reports`».
5. Переименуем каталог `reports/monthly.01` в `reports/monthly` командой «`mv reports/monthly.01 reports/monthly`».

```
jeffrey@jeffrey-VirtualBox:~$ touch may
jeffrey@jeffrey-VirtualBox:~$ ls -l may
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 17:31 may
jeffrey@jeffrey-VirtualBox:~$ chmod u+x may
jeffrey@jeffrey-VirtualBox:~$ ls -l may
-rwxrw-r-- 1 jeffrey jeffrey 0 мая 14 17:31 may
jeffrey@jeffrey-VirtualBox:~$ mkdir monthly
mkdir: cannot create directory 'monthly': File exists
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads      laboratory      monthly      Pictures      snap
Desktop   helloworld     laboratory2     Music        Public        Templates
Documents initial-commit  may            newdir       reports       Videos
jeffrey@jeffrey-VirtualBox:~$ rm -r abc1 monthly reports
jeffrey@jeffrey-VirtualBox:~$ ls
Desktop   helloworld     laboratory2     newdir       snap
Documents initial-commit  may            Pictures     Templates
Downloads laboratory      Music          Public       Videos
jeffrey@jeffrey-VirtualBox:~$ mkdir monthly
jeffrey@jeffrey-VirtualBox:~$
```

Выполним примеры, описанные в лабораторной работе


```

jeffrey@jeffrey-VirtualBox:~$ chmod g-r monthly; chmod o-r monthly
jeffrey@jeffrey-VirtualBox:~$ ls -l monthly
total 0
jeffrey@jeffrey-VirtualBox:~$ ls -l
total 60
drwxr-xr-x 3 jeffrey jeffrey 4096 мая  3 19:06 Desktop
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Documents
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Downloads
drwxrwxr-x 2 jeffrey jeffrey 4096 мая 14 16:45 helloworld
drwxrwxr-x 4 jeffrey jeffrey 4096 мая  3 20:30 initial-commit
drwxrwxr-x 3 jeffrey jeffrey 4096 мая  1 23:10 laboratory
drwxrwxr-x 4 jeffrey jeffrey 4096 мая  3 20:32 laboratory2
-rwxrw-r-- 1 jeffrey jeffrey   0 мая 14 17:31 may
drwx-wx--x 2 jeffrey jeffrey 4096 мая 14 17:33 monthly
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Music
drwxrwxr-x 2 jeffrey jeffrey 4096 мая 14 16:44 newdir
drwxr-xr-x 2 jeffrey jeffrey 4096 мая 14 17:33 Pictures
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Public
drwxr-xr-x 3 jeffrey jeffrey 4096 мая  3 19:02 snap
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Templates
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Videos
jeffrey@jeffrey-VirtualBox:~$ touch abc1
jeffrey@jeffrey-VirtualBox:~$ chmod g+w abc1
jeffrey@jeffrey-VirtualBox:~$ ls -l abc1
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 17:35 abc1
jeffrey@jeffrey-VirtualBox:~$

```

Выполним примеры, описанные в лабораторной работе

1. Выполняем следующие действия, отображенные на (рис. -fig. 3.5) (рис. -fig. 3.6) (рис. -fig. 3.7):
 2. Копируем файл /usr/include/sudo_plugin.h (т.к. у меня нет каталога /usr/include/sys/, то беру произвольный файл из каталога /usr/include/) в домашний каталог (команда «cp /usr/include/ sudo_plugin.h ~») и называем его equipment (команда «mv sudo_plugin.h equipment»).
 3. В домашнем каталоге создаем директорию ~/ski.places (команда «mkdir ski.places»).
1. Перемещаем файл equipment в каталог ~/ski.places (команда «mv equipment ski.places»).
 2. Переименовываем файл ~/ski.places/equipment в

~/ski.places/equiplist (команда «mv ski.places/equipment ski.places/equiplist»).

3. Создаем в домашнем каталоге файл abc1 (команда «touch abc1») и копируем его в каталог ~/ski.places (команда «cp abc1 ski.places»), называем его equiplist2 (команда «mv ski.places/abc1 ski.places/equiplist2»).
4. Создаем каталог с именем equipment в каталоге ~/ski.places (команда «mkdir ski.places/equipment»).
5. Перемещаем файлы ~/ski.places/equiplist и equiplist2 в каталог ~/ski.places/equipment (команда «mv ski.places/equiplist ski.places/equipment ski.places/equipment/equiplist2»).
6. Создаем (команда «mkdir newdir») и перемещаем каталог ~/newdir в каталог ~/ski.places (команда «mv newdir ski.places») и называем его plans (команда «mv ski.places/newdir ski.places/plans»).

```
jeffrey@jeffrey-VirtualBox:~$ cd /usr
jeffrey@jeffrey-VirtualBox:/usr$ cd include
jeffrey@jeffrey-VirtualBox:/usr/include$ ls
gcalc-2  gci-2  iproute2  openvpn  python3.8  reglib  sudo_plugin.h  X11  xorg
jeffrey@jeffrey-VirtualBox:/usr/include$ cp /usr/include/sudo_plugin.h ~
jeffrey@jeffrey-VirtualBox:/usr/include$
```

```
jeffrey@jeffrey-VirtualBox:/usr/include$ cp /usr/include/sudo_plugin.h ~
jeffrey@jeffrey-VirtualBox:/usr/include$ cd
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      file.txt      laboratory2  newdir      sudo_plugin.h
Desktop  helloworld    may         Pictures    Templates
Documents initial-commit monthly     Public      Videos
Downloads laboratory    Music       snap
jeffrey@jeffrey-VirtualBox:~$ mv sudo_plugin.h equipment
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  helloworld  laboratory2  Music      Public      Videos
Desktop  equipment  initial-commit  may         newdir     snap
Documents file.txt   laboratory    monthly     Pictures    Templates
jeffrey@jeffrey-VirtualBox:~$
```

Выполним действия из пункта 2 задания

```

jeffrey@jeffrey-VirtualBox:~$ mkdir ski.plases
jeffrey@jeffrey-VirtualBox:~$ mkdir equipment ski.plases
mkdir: cannot create directory 'equipment': File exists
mkdir: cannot create directory 'ski.plases': File exists
jeffrey@jeffrey-VirtualBox:~$ mv equipment ski.plases
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equipment
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  initial-commit  may      newdir     ski.plases  Videos
Desktop   file.txt   laboratory      monthly  Pictures   snap
Documents helloworld laboratory2     Music    Public     Templates
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/equipment ~/ski.plases/equiplist
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equiplist
jeffrey@jeffrey-VirtualBox:~$ touch abc1
jeffrey@jeffrey-VirtualBox:~$ cp abc ski.plases
cp: cannot stat 'abc': No such file or directory
jeffrey@jeffrey-VirtualBox:~$ cp abc1 ski.plases
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/abc1 ~/ski.plases/equiplist2
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equiplist  equiplist2
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  initial-commit  may      newdir     ski.plases  Videos
Desktop   file.txt   laboratory      monthly  Pictures   snap

```

Выполним действия из пункта 2 задания

```

jeffrey@jeffrey-VirtualBox:~$ mkdir ~/ski.plases/equipment
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equiplist  equiplist2  equipment
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/equiplis1^C
mv: missing destination file operand after '/home/jeffrey/ski.plases/equiplis1^
C'
Try 'mv --help' for more information.
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/equiplis1^C
mv: missing destination file operand after '/home/jeffrey/ski.plases/equiplis1^
C'
Try 'mv --help' for more information.
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equiplist  equiplist2  equipment
jeffrey@jeffrey-VirtualBox:~$ █

```



```
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/equiplist1 ~/ski.plases/equiplist
2 ~/ski.plases/equipment
mv: cannot stat '/home/jeffrey/ski.plases/equiplist1': No such file or director
y
jeffrey@jeffrey-VirtualBox:~$ mv ~/ski.plases/equiplist ~/ski.plases/equiplist2
~/ski.plases/equipment
mv: cannot stat '/home/jeffrey/ski.plases/equiplist2': No such file or director
y
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equipment
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases/equipment
equiplist  equiplist2
jeffrey@jeffrey-VirtualBox:~$ mkdir newdir
mkdir: cannot create directory 'newdir': File exists
jeffrey@jeffrey-VirtualBox:~$ mv newdir ski.plases
jeffrey@jeffrey-VirtualBox:~$ mv ski.plases/newdir ski.plases/plans
jeffrey@jeffrey-VirtualBox:~$ ls ski.plases
equipment  plans
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  initial-commit  may      Pictures  snap
Desktop   file.txt   laboratory      monthly  Public    Templates
Documents helloworld laboratory2     Music    ski.plases Videos
jeffrey@jeffrey-VirtualBox:~$
```

Выполним действия из пункта 2 задания

7. Определяем опции команды `chmod`, необходимые для того, чтобы присвоить соответствующим файлам выделенные права доступа, считая, что в начале таких прав нет. Предварительно создаем необходимые файлы, используя команды: «`mkdir australia play`», «`touch my_os feathers`».

- `drwxr-r- ... australia`: команда «`chmod 744 australia`» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только чтение)
- `drwx-x-x ... play`: команда «`chmod 711 play`» (это каталог, владелец имеет право на чтение, запись и выполнение, группа владельца и остальные – только выполнение)
- `-r-xr-r- ... my_os`: команда «`chmod 544 my_os`» (это файл, владелец имеет право на чтение и выполнение, группа владельца и остальные – только чтение)
- `-rw-rw-r- ... feathers`: команда «`chmod 664 feathers`» (это файл, владелец и группа владельца имеют право на чтение и запись, остальные – только чтение)

Командой «`ls -l`» проверяем правильность выполненных действий (рис. -fig. 3.8).

```
jeffrey@jeffrey-VirtualBox:~$ mkdir australia play
jeffrey@jeffrey-VirtualBox:~$ touch my_os feathers
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  initial-commit  monthly  play      Templates
australia feathers    laboratory      Music    Public    Videos
Desktop   file.txt   laboratory2     my_os    ski.plases
Documents helloworld may            Pictures snap
jeffrey@jeffrey-VirtualBox:~$ chmod 744 australia
jeffrey@jeffrey-VirtualBox:~$ chmod 711 play
jeffrey@jeffrey-VirtualBox:~$ chmod 544 my_os
jeffrey@jeffrey-VirtualBox:~$ chmod 664 feathers
jeffrey@jeffrey-VirtualBox:~$ ls -l
```

```

jeffrey@jeffrey-VirtualBox:~$ ls -l
total 72
-rw-rw-r-- 1 jeffrey jeffrey  0 мая 14 20:01 abc1
drwxr--r-- 2 jeffrey jeffrey 4096 мая 14 20:18 australia
drwxr-xr-x 3 jeffrey jeffrey 4096 мая  3 19:06 Desktop
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Documents
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Downloads
-rw-rw-r-- 1 jeffrey jeffrey  0 мая 14 20:18 feathers
-rw-rw-r-- 1 jeffrey jeffrey 2469 мая 14 17:54 file.txt
drwxrwxr-x 2 jeffrey jeffrey 4096 мая 14 16:45 helloworld
drwxrwxr-x 4 jeffrey jeffrey 4096 мая  3 20:30 initial-commit
drwxrwxr-x 3 jeffrey jeffrey 4096 мая  1 23:10 laboratory
drwxrwxr-x 3 jeffrey jeffrey 4096 мая 14 18:46 laboratory2
-rw-rw-r-- 1 jeffrey jeffrey  0 мая 14 19:02 may
drwx-wx--x 2 jeffrey jeffrey 4096 мая 14 19:03 monthly
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Music
-r-xr--r-- 1 jeffrey jeffrey  0 мая 14 20:18 my_os
drwxr-xr-x 2 jeffrey jeffrey 4096 мая 14 20:17 Pictures
drwx--x--x 2 jeffrey jeffrey 4096 мая 14 20:18 play
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Public
drwxrwxr-x 4 jeffrey jeffrey 4096 мая 14 20:16 ski.places
drwxr-xr-x 3 jeffrey jeffrey 4096 мая  3 19:02 snap
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Templates
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Videos
jeffrey@jeffrey-VirtualBox:~$

```

Определяем опции команды chmod

Выполняем следующие действия, отображенные на (рис. -fig. 3.9) (рис.-fig. 3.10) (рис. -fig. 3.11):

- 1.Просмотрим содержимое файла /etc/passwd (команда «cat /etc/passwd»).
- 2.Копируем файл ~/feathers в файл ~/file.old (команда «cp feathers file.old»).
- 3.Переместим файл ~/file.old в каталог ~/play (команда «mv file.ord play»).
- 4.Скопируем каталог ~/play в каталог ~/fun (команда «cp -r play fun»).
- 5.Переместим каталог ~/fun в каталог ~/play (команда «mv fun play») и назовем его games (команда «mv play/fun play/games»).
- 6.Лишим владельца файла ~/feathers права на чтение (команда «chmodu- r feathers»).
- 7.Если мы попытаемся просмотреть файл ~/feathers командой cat, то получим отказ в доступе, т.к. в предыдущем пункте лишили владельца права на чтение данного файла.
- 8.Если мы попытаемся скопировать файл ~/feathers, например, в каталог monthly, то получим отказ в доступе, по причине, описанной в предыдущем пункте.
- 9.Дадим владельцу файла ~/feathers право на чтение (команда «chmod u+r feathers»).
- 10.Лишим владельца каталога ~/play права на выполнение (команда «chmod u-x play»).

11.Перейдем в каталог ~/play (команда «cd play»). Получим отказ в доступе, т.к. в предыдущем пункте лишили владельца права на выполнение данного каталога.

12.Дадим владельцу каталога ~/play право на выполнение (команда «chmod u+x play»).

```
jeffrey@jeffrey-VirtualBox:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nolog
in
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/
nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sb
in/nologin
messagebus:x:103:106:./nonexistent:/usr/sbin/nologin
syslog:x:104:110:./home/syslog:/usr/sbin/nologin
_apt:x:105:65534:./nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
```

Просмотрим содержимое файла /etc/passwd


```

jeffrey@jeffrey-VirtualBox:~$ cp feathers file.old
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  helloworld  may      Pictures  snap
australia feathers    initial-commit monthly  play      Templates
Desktop   file.old   laboratory   Music    Public    Videos
Documents file.txt   laboratory2  my_os    ski.plases
jeffrey@jeffrey-VirtualBox:~$ mv file.old play
jeffrey@jeffrey-VirtualBox:~$ ls
abc1      Downloads  initial-commit  monthly  play      Templates
australia feathers    laboratory       Music    Public    Videos
Desktop   file.txt   laboratory2      my_os    ski.plases
Documents helloworld may              Pictures  snap
jeffrey@jeffrey-VirtualBox:~$ ls play
file.old
jeffrey@jeffrey-VirtualBox:~$ mkdir fun
jeffrey@jeffrey-VirtualBox:~$ cp -r play fun
jeffrey@jeffrey-VirtualBox:~$ ls fun
play
jeffrey@jeffrey-VirtualBox:~$ mv fun play
jeffrey@jeffrey-VirtualBox:~$ mv play/fun play/games
jeffrey@jeffrey-VirtualBox:~$ ls play
file.old  games
jeffrey@jeffrey-VirtualBox:~$ ls play/games
play
jeffrey@jeffrey-VirtualBox:~$ chmod u-r feathers
jeffrey@jeffrey-VirtualBox:~$ ls -l feathers
--w-rw-r-- 1 jeffrey jeffrey 0 мая 14 20:18 feathers
jeffrey@jeffrey-VirtualBox:~$ cat feathers
cat: feathers: Permission denied
jeffrey@jeffrey-VirtualBox:~$ chmod u+r feathers
jeffrey@jeffrey-VirtualBox:~$ ls -l feathers
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 20:18 feathers
jeffrey@jeffrey-VirtualBox:~$ chmod u-x play
jeffrey@jeffrey-VirtualBox:~$ ls -l play
ls: cannot access 'play/file.old': Permission denied
ls: cannot access 'play/games': Permission denied
total 0
-????????? ? ? ? ?      ? file.old
d????????? ? ? ? ?      ? games
jeffrey@jeffrey-VirtualBox:~$

```

Выполним действия из пункта 3 задания


```

jeffrey@jeffrey-VirtualBox:~$ chmod u+x play
jeffrey@jeffrey-VirtualBox:~$ ls -l play
total 4
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 20:21 file.old
drwxrwxr-x 3 jeffrey jeffrey 4096 мая 14 20:22 games
jeffrey@jeffrey-VirtualBox:~$ ls -l
total 72
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 20:01 abc1
drwxr--r-- 2 jeffrey jeffrey 4096 мая 14 20:18 australia
drwxr-xr-x 3 jeffrey jeffrey 4096 мая 3 19:06 Desktop
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Documents
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Downloads
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 20:18 feathers
-rw-rw-r-- 1 jeffrey jeffrey 2469 мая 14 17:54 file.txt
drwxrwxr-x 2 jeffrey jeffrey 4096 мая 14 16:45 helloworld
drwxrwxr-x 4 jeffrey jeffrey 4096 мая 3 20:30 initial-commit
drwxrwxr-x 3 jeffrey jeffrey 4096 мая 1 23:10 laboratory
drwxrwxr-x 3 jeffrey jeffrey 4096 мая 14 18:46 laboratory2
-rw-rw-r-- 1 jeffrey jeffrey 0 мая 14 19:02 may
drwx-wx--x 2 jeffrey jeffrey 4096 мая 14 19:03 monthly
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Music
-r-xr--r-- 1 jeffrey jeffrey 0 мая 14 20:18 my_os
drwxr-xr-x 2 jeffrey jeffrey 4096 мая 14 20:26 Pictures
drwx--x--x 3 jeffrey jeffrey 4096 мая 14 20:22 play
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Public
drwxrwxr-x 4 jeffrey jeffrey 4096 мая 14 20:16 ski.places
drwxr-xr-x 3 jeffrey jeffrey 4096 мая 3 19:02 snap
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Templates
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Videos

```

Выполним действия из пункта 4 задания

```

drwxr-xr-x 3 jeffrey jeffrey 4096 мая 3 19:02 snap
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Templates
drwxr-xr-x 2 jeffrey jeffrey 4096 апр 23 17:02 Videos
jeffrey@jeffrey-VirtualBox:~$ man mount
jeffrey@jeffrey-VirtualBox:~$ █

```

8. Используя команды «man mount», «man fsck», «man mkfs», «man kill», получим информацию о соответствующих командах.

Команда mount (рис. 3.12):

Предназначена для монтирования файловой системы. Все файлы, доступные в Unix системах, составляют иерархическую файловую структуру, которая имеет ветки (каталоги) и листья (файлы в каталогах). Корень этого дерева обозначается как /. Физически файлы могут располагаться на различных устройствах. Команда mount служит для подключения файловых систем разных устройств к этому большому дереву.

Наиболее часто встречающаяся форма команды mount выглядит следующим образом: «mount -t vfstype devicedir» Такая команда предлагает ядру смонтировать (подключить) файловую систему указанного типа vfstype, расположенную на устройстве device, к заданному каталогу dir, который часто называют точкой монтирования.

```
MOUNT(8)                                System Administration                                MOUNT(8)

NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-l|-h|-V]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|dir

    mount [-fnrsvw] [-t fstype] [-o options] device dir

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree,
    the file hierarchy, rooted at /. These files can be spread out over
    several devices. The mount command serves to attach the filesystem
    found on some device to the big file tree. Conversely, the umount(8)
    command will detach it again. The filesystem is used to control how
    data is stored on the device or provided in a virtual way by network
    or another services.

    The standard form of the mount command is:

        mount -t type device dir

    This tells the kernel to attach the filesystem found on device (which
Manual page mount(8) line 1 (press h for help or q to quit)
```

Рис. 3.12: Команда mount

9. Команда fsck (рис. -fig. 3.13):

Это утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых системах Linux. Он использует программы, специфичные для типа файловой системы, которую он проверяет.

У команды fsck следующий синтаксис:

fsck параметр – параметры ФС...

Например, если нужно восстановить («починить») файловую систему на некотором устройстве /dev/sdb2, следует воспользоваться командой:

«sudo fsck -y /dev/sdb2»

Опция -y необходима, т. к. при её отсутствии придётся слишком часто давать подтверждение.

```
FCK(8)                                System Administration                                FCK(8)

NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--]
    [fs-specific-options]

DESCRIPTION
    fsck is used to check and optionally repair one or more Linux filesystems.
    filesystem can be a device name (e.g. /dev/hdc1, /dev/sdb2), a mount point
    (e.g. /, /usr, /home), or an filesystem label or UUID specifier (e.g.
    UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root). Normally, the
    fsck program will try to handle filesystems on different physical disk
    drives in parallel to reduce the total amount of time needed to check all
    of them.

    If no filesystems are specified on the command line, and the -A option is
    not specified, fsck will default to checking filesystems in /etc/fstab
    serially. This is equivalent to the -As options.

    The exit code returned by fsck is the sum of the following conditions:

        0      No errors
        1      Filesystem errors corrected
        2      System should be rebooted
        4      Filesystem errors left uncorrected

Manual page fsck(8) line 1 (press h for help or q to quit)
```

Рис. 3.13: Команда fsck

10. Команда mkfs (рис. -fig. 3.14):

Создаёт новую файловую систему Linux.

Имеет следующий синтаксис:

mkfs -V -t fstype fs-options filesystem blocks

mkfs используется для создания файловой системы Linux на некотором устройстве, обычно в разделе жёсткого диска. В качестве аргумента filesystem для файловой системы может выступать или название устройства (например, /dev/hda1, /dev/sdb2) или точка монтирования (например, /, /usr, /home).

Аргументом blocks указывается количество блоков, которые выделяются для использования этой файловой системой.

По окончании работы mkfs возвращает 0 - в случае успеха, а 1 - при неудачной операции.

Например, команда «mkfs -t ext2 /dev/hdb1» создаёт файловую систему типа ext2 в разделе /dev/hdb1 (второй жёсткий диск).

```
MKFS(8)                               System Administration                               MKFS(8)

NAME
    mkfs - build a Linux filesystem

SYNOPSIS
    mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
    This mkfs frontend is deprecated in favour of filesystem specific
    mkfs.<type> utils.

    mkfs is used to build a Linux filesystem on a device, usually a hard
    disk partition. The device argument is either the device name (e.g.
    /dev/hda1, /dev/sdb2), or a regular file that shall contain the
    filesystem. The size argument is the number of blocks to be used for
    the filesystem.

    The exit code returned by mkfs is 0 on success and 1 on failure.

    In actuality, mkfs is simply a front-end for the various filesystem
    builders (mkfs.fstype) available under Linux. The filesystem-specific
    builder is searched for via your PATH environment setting only.
    Please see the filesystem-specific builder manual pages for further
    details.

OPTIONS
    -t, --type type
Manual page mkfs(8) line 1 (press h for help or q to quit)
```

Рис. 3.14: Команда mkfs

11. Команда kill (рис. -fig. 3.15):

Посылает сигнал процессу или выводит список допустимых сигналов. Имеет следующий синтаксис:

kill опции PID, где PID – это PID (числовой идентификатор) процесса или несколько PID процессов, если требуется послать сигнал сразу нескольким процессам. Например, команда «kill -KILL 3121» посылает сигнал KILL процессу с PID 3121, чтобы принудительно завершить процесс.

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

NAME
    kill - send a signal to a process

SYNOPSIS
    kill [options] <pid> [...]

DESCRIPTION
    The default signal for kill is TERM. Use -l or -L to list available
    signals. Particularly useful signals include HUP, INT, KILL, STOP,
    CONT, and 0. Alternate signals may be specified in three ways: -9,
    -SIGKILL or -KILL. Negative PID values may be used to choose whole
    process groups; see the PGID column in ps command output. A PID of -1
    is special; it indicates all processes except the kill process itself
    and init.

OPTIONS
    <pid> [...]
        Send signal to every <pid> listed.

    -<signal>
    -s <signal>
    --signal <signal>
        Specify the signal to be sent. The signal can be specified by
        using name or number. The behavior of signals is explained in
        signal(7) manual page.

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

Рис. 3.15: Команда kill

12. Контрольные вопросы

13. Чтобы узнать, какие файловые системы существуют на жёстком диске моего компьютера, использую команду «df -Th». На моем компьютере есть следующие файловые системы: devtmpfs, tmpfs, ext4, iso9660. devtmpfs позволяет ядру создать экземпляр tmpfs с именем devtmpfs при инициализации ядра, прежде чем регистрируется какое-либо устройство с драйверами. Каждое устройство с майором / минором будет предоставлять узел устройства в devtmpfs. devtmpfs монтируется на /dev и содержит специальные файлы устройств для всех устройств. tmpfs – временное файловое хранилище во многих Unix-подобных ОС. Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо ПЗУ. Подобная конструкция является RAM диском. Данная файловая система также предназначена для быстрого и ненадёжного хранения временных данных. Хорошо подходит для /tmp и массовой сборки пакетов/образов. Предполагает наличие достаточного объёма виртуальной памяти. Файловая система tmpfs предназначена для того, чтобы использовать часть физической памяти сервера как обычный дисковый раздел, в котором можно сохранять данные (чтение и запись). Поскольку данные размещены в памяти, то чтение или запись происходят много раз быстрее, чем с обычного HDD диска. ext4 – имеет обратную совместимость с предыдущими версиями ФС. Эта версия была выпущена в 2008 году. Является первой ФС из «семейства» Ext, использующая механизм

«extent file system», который позволяет добиться меньшей фрагментации файлов и увеличить общую производительность файловой системы. Кроме

того, в Ext4 реализован механизм отложенной записи (delayed allocation – delalloc), который так же уменьшает фрагментацию диска и снижает нагрузку на CPU. С другой стороны, хотя механизм отложенной записи и используется во многих ФС, но в силу сложности своей реализации он повышает вероятность утери данных.

Характеристики:

- максимальный размер файла: 16 TB;
- максимальный размер раздела: 16 TB;
- максимальный размер имени файла: 255 символов.

Рекомендации по использованию:

- наилучший выбор для SSD;
- наилучшая производительность по сравнению с предыдущими Ext-системами;
- она так же отлично подходит в качестве файловой системы для серверов баз данных, хотя сама система и моложе Ext3.

ISO 9660 – стандарт, выпущенный Международной организацией по стандартизации, описывающий файловую систему для дисков CD-ROM. Также известен как CDFS (Compact Disc File System). Целью стандарта является обеспечить совместимость носителей под разными операционными системами, такими, как Unix, Mac OS, Windows.

10. Файловая система Linux/UNIX физически представляет собой пространство раздела диска, разбитое на блоки фиксированного размера, кратные размеру сектора – 1024, 2048, 4096 или 8120 байт. Размер блока указывается при создании файловой системы. В файловой структуре Linux имеется один корневой раздел – / (он же root, корень). Все разделы жесткого диска (если их несколько) представляют собой структуру подкаталогов, “примонтированных” к определенным каталогам.

- / – корень

Это главный каталог в системе Linux. По сути, это и есть файловая система Linux. Адреса всех файлов начинаются с корня, а дополнительные разделы, флешки или оптические диски подключаются в папки корневого каталога. Только пользователь root имеет право читать и изменять файлы в этом каталоге.

- /BIN – бинарные файлы пользователя

Этот каталог содержит исполняемые файлы. Здесь расположены программы, которые можно использовать в однопользовательском режиме или режиме восстановления.

- /SBIN – системные исполняемые файлы

Так же, как и /bin, содержит двоичные исполняемые файлы, которые доступны на ранних этапах загрузки, когда не примонтирован каталог /usr. Но здесь находятся программы, которые можно выполнять только с правами суперпользователя.

- /ETC – конфигурационные файлы

В этой папке содержатся конфигурационные файлы всех программ, установленных в системе. Кроме конфигурационных файлов, в системе инициализации Init Scripts, здесь находятся скрипты запуска и завершения системных демонов, монтирования файловых систем и автозагрузки программ.

- /DEV – файлы устройств

В Linux все, в том числе внешние устройства являются файлами. Таким образом, все подключенные флешки, клавиатуры, микрофоны, камеры — это просто файлы в каталоге /dev/. Выполняется сканирование всех подключенных устройств и создание для них специальных файлов.

- /PROC – информация о процессах

По сути, это псевдофайловая система, содержащая подробную информацию о каждом процессе, его Pid, имя исполняемого файла, параметры запуска, доступ к оперативной памяти и так далее. Также здесь можно найти

информацию об использовании системных ресурсов.

- **/VAR** – переменные файлы

Название каталога `/var` говорит само за себя, он должен содержать файлы, которые часто изменяются. Размер этих файлов постоянно увеличивается. Здесь содержатся файлы системных журналов, различные кешы, базы данных и так далее.

- **/TMP** – временные файлы

В этом каталоге содержатся временные файлы, созданные системой, любыми программами или пользователями. Все пользователи имеют право записи в эту директорию.

- **/USR** – программы пользователя

Это самый большой каталог с большим количеством функций. Здесь находятся исполняемые файлы, исходники программ, различные ресурсы приложений, картинки, музыку и документацию.

- **/HOME** – домашняя папка

В этой папке хранятся домашние каталоги всех пользователей. В них они могут хранить свои личные файлы, настройки программ и т. д.

- **/BOOT** – файлы загрузчика

Содержит все файлы, связанные с загрузчиком системы. Это ядро `vmlinuz`, образ `initrd`, а также файлы загрузчика, находящиеся в каталоге `/boot/grub`.

- **/LIB** – системные библиотеки

Содержит файлы системных библиотек, которые используются исполняемыми файлами в каталогах `/bin` и `/sbin`.

- **/OPT** – дополнительные программы

В эту папку устанавливаются проприетарные программы, игры или драйвера. Это программы, созданные в виде отдельных исполняемых файлов самими производителями.

- **/MNT** – монтирование

В этот каталог системные администраторы могут монтировать внешние

или дополнительные файловые системы.

- /MEDIA – съемные носители

В этот каталог система монтирует все подключаемые внешние накопители –USB флешки, оптические диски и другие носители информации.

- /SRV – сервер. В этом каталоге содержатся файлы серверов и сервисов.
- /RUN - процессы Каталог, содержащий PID файлы процессов, похожий на /var/run, но в отличие от него, он размещен в TMPFS, а поэтому после перезагрузки все файлы теряются.

11. Чтобы содержимое некоторой файловой системы было доступно операционной системе необходимо воспользоваться командой mount.

12. Целостность файловой системы может быть нарушена из-за перебоев в питании, неполадок в оборудовании или из-за некорректного/внезапного выключения компьютера. Чтобы устранить повреждения файловой системы необходимо использовать команду fsck.

13. Файловую систему можно создать, используя команду mkfs. Ее краткое описание дано в пункте 5 в ходе выполнения заданий лабораторной работы.

14. Для просмотра текстовых файлов существуют следующие команды:

- cat

Задача команды cat очень проста – она читает данные из файла или стандартного ввода и выводит их на экран.

Синтаксис утилиты:

cat опции файл1 файл2 ...

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

- nl

Команда nl действует аналогично команде cat, но выводит еще и номера строк в столбце слева.

- less

Существенно более развитая команда для пролистывания текста.

При чтении данных со стандартного ввода она создает буфер, который позволяет листать текст как вперед, так и назад, а также искать как по направлению к концу, так и по направлению к началу текста.

Синтаксис аналогичный синтаксису команды cat.

Некоторые опции:

-g – при поиске подсвечивать только текущее найденное слово (по умолчанию подсвечиваются все вхождения)

-N – показывать номера строк

- 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 аналогичный синтаксису команды cat.

Основные опции:

-c (–bytes) – позволяет задавать количество текста не в строках, а в байтах

-n (–lines) – показывает заданное количество строк вместо 10, которые выводятся по умолчанию

-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

- tail

Эта команда позволяет выводить заданное количество строк с конца файла, а также выводить новые строки в интерактивном режиме.

Синтаксис аналогичный синтаксису команды cat.

Основные опции:

- 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

15. Утилита `cp` позволяет полностью копировать файлы и директории. Синтаксис:

`cp` опции файл-источник файл-приемник

После выполнения команды файл-источник будет полностью перенесен в файл-приемник. Если в конце указан слэш, файл будет записан в заданную директорию с оригинальным именем.

Основные опции:

- attributes-only – не копировать содержимое файла, а только флаги доступа и владельца
- f, –force – перезаписывать существующие файлы
- i, –interactive – спрашивать, нужно ли перезаписывать существующие файлы
- L – копировать не символические ссылки, а то, на что они указывают -n – не перезаписывать существующие файлы
- P – не следовать символическим ссылкам
- r – копировать папку Linux рекурсивно
- s – не выполнять копирование файлов в Linux, а создавать символические

ссылки

- u – скопировать файл, только если он был изменён
- x – не выходить за пределы этой файловой системы
- r – сохранять владельца, временные метки и флаги доступа при копировании
- t – считать файл-приемник директорией и копировать файл-источник в эту директорию

16. Команда `mv` используется для перемещения одного или нескольких файлов (или директорий) в другую директорию, а также для переименования файлов и директорий.

Синтаксис:

`mv` -опции старый_файл новый_файл

Основные опции:

- help – выводит на экран официальную документацию об утилите
- version – отображает версию `mv`
- b – создает копию файлов, которые были перемещены или перезаписаны
- f – при активации не будет спрашивать разрешение у владельца файла, если речь идет о перемещении или переименовании файла -i – наоборот, будет спрашивать разрешение у владельца
- n – отключает перезапись существующих объектов
- strip-trailing-slashes — удаляет завершающий символ / у файла при его наличии
- t директория — перемещает все файлы в указанную директорию
- u – осуществляет перемещение только в том случае, если исходный файл новее объекта назначения
- v – отображает сведения о каждом элементе во время обработки команды

Команда `rename` также предназначена, чтобы переименовать файл.

Синтаксис:

`rename` опции старое_имя новое_имя файлы

Основные опции:

-v – вывести список обработанных файлов

-n – тестовый режим, на самом деле никакие действия выполнены не будут

-f – принудительно перезаписывать существующие файлы

17. Права доступа – совокупность правил, регламентирующих порядок и условия доступа субъекта к объектам информационной системы (информации, её носителям, процессам и другим ресурсам) установленных правовыми документами или собственником, владельцем информации.

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой `chmod`. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.

Синтаксис команды:

`chmod режим имя_файла`

Режим имеет следующие компоненты структуры и способ записи:

- = установить право
- – лишить права
- + дать право
- r чтение
- w запись
- x выполнение
- u (user) владелец файла
- g (group) группа, к которой принадлежит владелец файла
- o (others) все остальные

14. Выводы

В ходе выполнения данной лабораторной работы я ознакомился с файловой системой Linux, её структурой, именами и содержанием каталогов, получил навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.