

Отчет к лабораторной работе №5

Дисциплина: Операционные системы

Маслова Анастасия, НКНбд-01-21

Содержание

Цель работы	1
Задание	1
Теоретическое введение	2
Команды для работы с файлами и каталогами	2
Копирование файлов и каталогов	3
Перемещение и переименование файлов и каталогов.....	3
Выполнение лабораторной работы	4
Выводы	12
Список литературы	12

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.

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: 2.1. 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. 2.2. В домашнем каталоге создайте директорию `~/ski.places`. 2.3. Переместите файл `equipment` в каталог `~/ski.places`. 2.4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`. 2.5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`. 2.6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`. 2.7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`. 2.8. 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: 3.1. `drwxr-g-` ... `australia` 3.2. `drwx-x-x` ... `play` 3.3. `-r-xr-g-` ... `my_os` 3.4. `-rw-rw-r-` ... `feathers` При необходимости создайте нужные файлы.
4. Прделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды: 4.1. Просмотрите содержимое файла `/etc/passwd`. 4.2. Скопируйте файл `~/feathers` в файл `~/file.old`. 4.3. Переместите файл `~/file.old` в каталог `~/play`. 4.4. Скопируйте каталог `~/play` в каталог `~/fun`. 4.5. Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`. 4.6. Лишите владельца файла `~/feathers` права на чтение. 4.7. Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`? 4.8. Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`? 4.9. Дайте владельцу файла `~/feathers` право на чтение. 4.10. Лишите владельца каталога `~/play` права на выполнение. 4.11. Перейдите в каталог `~/play`. Что произошло? 4.12. Дайте владельцу каталога `~/play` право на выполнение.
5. Прочитайте ман по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

Теоретическое введение

Команды для работы с файлами и каталогами

Для создания текстового файла можно использовать команду `touch`. Формат команды:

`touch` имя-файла

Для просмотра файлов небольшого размера можно использовать команду `cat`. Формат команды:

`cat` имя-файла

Для просмотра файлов постранично удобнее использовать команду `less`. Формат команды:

`less` имя-файла

Следующие клавиши используются для управления процессом просмотра: `-` Space — переход к следующей странице, `-` ENTER — сдвиг вперёд на одну строку, `-` b — возврат на предыдущую страницу, `-` h — обращение за подсказкой, `-` q — выход из режима просмотра файла.

Команда `head` выводит по умолчанию первые 10 строк файла. Формат команды:

`head` [-n] имя-файла,

где `n` — количество выводимых строк.

Команда `tail` выводит по умолчанию 10 последних строк файла. Формат команды:

`tail` [-n] имя-файла,

где n — количество выводимых строк.

Копирование файлов и каталогов

Команда `cp` используется для копирования файлов и каталогов. Формат команды:

```
cp [-опции] исходный_файл целевой_файл
```

Примеры: 1. Копирование файла в текущем каталоге. Скопировать файл `~/abc1` в файл `april` и в файл `may`:

```
cd
touch abc1
cp abc1 april
cp abc1 may
```

2. Копирование нескольких файлов в каталог. Скопировать файлы `april` и `may` в каталог `monthly`:

```
mkdir monthly
cp april may monthly
```

3. Копирование файлов в произвольном каталоге. Скопировать файл `monthly/may` в файл с именем `june`:

```
cp monthly/may monthly/june
ls monthly
```

Опция `i` в команде `cp` выведет на экран запрос подтверждения о перезаписи файла. Для рекурсивного копирования каталогов, содержащих файлы, используется команда `cp` с опцией `r`. Примеры: 1. Копирование каталогов в текущем каталоге. Скопировать каталог `monthly` в каталог `monthly.00`:

```
mkdir monthly.00
cp -r monthly monthly.00
```

2. Копирование каталогов в произвольном каталоге. Скопировать каталог `monthly.00` в каталог `/tmp`

```
cp -r monthly.00 /tmp
```

Перемещение и переименование файлов и каталогов

Команды `mv` и `mvdir` предназначены для перемещения и переименования файлов и каталогов. Формат команды `mv`:

```
mv [-опции] старый_файл новый_файл
```

Примеры: 1. Переименование файлов в текущем каталоге. Изменить название файла `april` на `july` в домашнем каталоге:

```
cd
mv april july
```

2. Перемещение файлов в другой каталог. Переместить файл `july` в каталог `monthly.00`:

```
mv july monthly.00
ls monthly.00
```

Результат:

```
april july june may
```

Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию `i`.

3. Переименование каталогов в текущем каталоге. Переименовать каталог `monthly.00` в `monthly.01`

```
mv monthly.00 monthly.01
```

4. Перемещение каталога в другой каталог. Переместить каталог `monthly.01` в каталог `reports`:

```
mkdir reports
```

```
mv monthly.01 reports
```

5. Переименование каталога, не являющегося текущим. Переименовать каталог `reports/monthly.01` в `reports/monthly`:

```
mv reports/monthly.01 reports/monthly
```

Полное теоретическую справку смотреть [здесь](#)

Выполнение лабораторной работы

1. Сначала я выполнила все примеры, представленные в теоретическом введении лабораторной работы.
2. Далее я создала файл `equipment`. В домашнем каталоге создала директорию `~/ski.plases`, переместила файл `equipment` в каталог `~/ski.plases`, переименовала файл `~/ski.plases/equipment` в `~/ski.plases/equiplist`, создала в домашнем каталоге файл `abc1` и скопировала его в каталог `~/ski.plases`, назвав его `equiplist2`, далее я создала каталог с именем `equipment` в каталоге `~/ski.plases`, переместила файлы `~/ski.plases/equiplist` и `equiplist2` в каталог `~/ski.plases/equipment`, создала и переместила каталог `~/newdir` в каталог `~/ski.plases` и назвала его `plans`. (рис.1)

```

[asmaslova@fedora ~]$ cd
[asmaslova@fedora ~]$ touch task
[asmaslova@fedora ~]$ ls -l task
-rw-rw-r--. 1 asmaslova asmaslova 0 May  7 15:42 task
[asmaslova@fedora ~]$ chmod u+x task
[asmaslova@fedora ~]$ ls -l task
-rwxrw-r--. 1 asmaslova asmaslova 0 May  7 15:42 task
[asmaslova@fedora ~]$ chmod u-x task
[asmaslova@fedora ~]$ ls -l task
-rw-rw-r--. 1 asmaslova asmaslova 0 May  7 15:42 task
[asmaslova@fedora ~]$ ls
2          3-220  Documents  example  memos  Music  Public  task  Videos
2-2.txt  Desktop  Downloads  letters  misk  Pictures  stack.yaml  Templates  work
[asmaslova@fedora ~]$ chmod g-r, o-r 3-220/
chmod: invalid mode: 'g-r,'
Try 'chmod --help' for more information.
[asmaslova@fedora ~]$ chmod g-r , o-r 3-220/
chmod: cannot access ',': No such file or directory
chmod: cannot access 'o-r': No such file or directory
[asmaslova@fedora ~]$ chmod g-r 3-220/
[asmaslova@fedora ~]$ chmod o-r 3-220/
[asmaslova@fedora ~]$ ls -l 3-220/
total 4
-rw-rw-r--. 1 asmaslova asmaslova  0 May  7 15:29 1-1.txt
-rw-rw-r--. 1 asmaslova asmaslova  0 May  7 15:29 2-2.txt
-rw-rw-r--. 1 asmaslova asmaslova 599 May  7 15:29 example.txt

```

рис.1 Выполнение задания второго пункта

3. Далее я создала каталоги australia, play и файлы my_os, feathers, над которыми проводила указанные в задании манипуляции с правами доступа (рис.2-4)

```

[asmaslova@fedora ski.plases]$ touch australia play my_os feathers
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:15 ./
drwx----- 1 asmaslova asmaslova 696 May 7 16:14 ../
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 australia
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 feathers
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 play
[asmaslova@fedora ski.plases]$ chmod o+r australia
[asmaslova@fedora ski.plases]$ chmod o+w australia
[asmaslova@fedora ski.plases]$ chmod o+x australia
[asmaslova@fedora ski.plases]$ chmod o-w australia
[asmaslova@fedora ski.plases]$ chmod -+x australia
[asmaslova@fedora ski.plases]$ chmod o-x australia
[asmaslova@fedora ski.plases]$ chmod u+r australia
[asmaslova@fedora ski.plases]$ chmod u+w australia
[asmaslova@fedora ski.plases]$ chmod u+x australia
[asmaslova@fedora ski.plases]$ chmod g+r australia
[asmaslova@fedora ski.plases]$ chmod g-w australia
[asmaslova@fedora ski.plases]$ chmod g-x australia
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:15 ./
drwx----- 1 asmaslova asmaslova 696 May 7 16:14 ../
-rwxr--r-- 1 asmaslova asmaslova 0 May 7 16:15 australia*
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 feathers
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 play
[asmaslova@fedora ski.plases]$ rm australia play
[asmaslova@fedora ski.plases]$ mkdir australia play
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:20 ./
drwx----- 1 asmaslova asmaslova 696 May 7 16:14 ../
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:20 australia/
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 feathers
-rw-rw-r-- 1 asmaslova asmaslova 0 May 7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:20 play/

```

рис.2. Создание необходимых каталогов и файлов и проверка их прав доступа


```

[asmaslova@fedora ski.plases]$ chmod u+r australia
[asmaslova@fedora ski.plases]$ chmod u+w australia
[asmaslova@fedora ski.plases]$ chmod u+x australia
[asmaslova@fedora ski.plases]$ chmod g+r australia
[asmaslova@fedora ski.plases]$ chmod g-w australia
[asmaslova@fedora ski.plases]$ chmod g-x australia
[asmaslova@fedora ski.plases]$ chmod o+r australia
[asmaslova@fedora ski.plases]$ chmod o-w australia
[asmaslova@fedora ski.plases]$ chmod o-x australia
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:20 ./
drwx-----. 1 asmaslova asmaslova 696 May 7 16:14 ../
drwxr--r--. 1 asmaslova asmaslova 0 May 7 16:20 australia/
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 feathers
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:20 play/
[asmaslova@fedora ski.plases]$ chmod u+r play
[asmaslova@fedora ski.plases]$ chmod u+w play
[asmaslova@fedora ski.plases]$ chmod u+x play
[asmaslova@fedora ski.plases]$ chmod g+x play
[asmaslova@fedora ski.plases]$ chmod g-r play
[asmaslova@fedora ski.plases]$ chmod g-w play
[asmaslova@fedora ski.plases]$ chmod o+x play
[asmaslova@fedora ski.plases]$ chmod o-r play
[asmaslova@fedora ski.plases]$ chmod o-w play
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:20 ./
drwx-----. 1 asmaslova asmaslova 696 May 7 16:14 ../
drwxr--r--. 1 asmaslova asmaslova 0 May 7 16:20 australia/
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 feathers
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
drwx--x--x. 1 asmaslova asmaslova 0 May 7 16:20 play/

```

рис.3 Настройка прав доступа для директив

```

[asmaslova@fedora ski.plases]$ chmod u+r my_os
[asmaslova@fedora ski.plases]$ chmod u+x my_os
[asmaslova@fedora ski.plases]$ chmod u-w my_os
[asmaslova@fedora ski.plases]$ chmod g+r my_os
[asmaslova@fedora ski.plases]$ chmod o+r my_os
[asmaslova@fedora ski.plases]$ chmod o-w my_os
[asmaslova@fedora ski.plases]$ chmod o-x my_os
[asmaslova@fedora ski.plases]$ chmod g-x my_os
[asmaslova@fedora ski.plases]$ chmod g-w my_os
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:20 ./
drwx-----. 1 asmaslova asmaslova 696 May 7 16:14 ../
drwxr--r--. 1 asmaslova asmaslova 0 May 7 16:20 australia/
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 feathers
-r-xr--r--. 1 asmaslova asmaslova 0 May 7 16:15 my_os*
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
drwx--x--x. 1 asmaslova asmaslova 0 May 7 16:20 play/
[asmaslova@fedora ski.plases]$ chmod u+r feathers
[asmaslova@fedora ski.plases]$ chmod u+w feathers
[asmaslova@fedora ski.plases]$ chmod u-x feathers
[asmaslova@fedora ski.plases]$ chmod g+r feathers
[asmaslova@fedora ski.plases]$ chmod g+w feathers
[asmaslova@fedora ski.plases]$ chmod g-x feathers
[asmaslova@fedora ski.plases]$ chmod o+r feathers
[asmaslova@fedora ski.plases]$ chmod o-w feathers
[asmaslova@fedora ski.plases]$ chmod o-x feathers
[asmaslova@fedora ski.plases]$ ls -alF
total 0
drwxrwxr-x. 1 asmaslova asmaslova 80 May 7 16:20 ./
drwx-----. 1 asmaslova asmaslova 696 May 7 16:14 ../
drwxr--r--. 1 asmaslova asmaslova 0 May 7 16:20 australia/
drwxrwxr-x. 1 asmaslova asmaslova 38 May 7 16:13 equipment/
-rw-rw-r--. 1 asmaslova asmaslova 0 May 7 16:15 feathers
-r-xr--r--. 1 asmaslova asmaslova 0 May 7 16:15 my_os*
drwxrwxr-x. 1 asmaslova asmaslova 0 May 7 16:14 plans/
drwx--x--x. 1 asmaslova asmaslova 0 May 7 16:20 play/
[asmaslova@fedora ski.plases]$ cd
[asmaslova@fedora ~]$ cat ~/etc/password
cat: /home/asmaslova/etc/password: No such file or directory
[asmaslova@fedora ~]$ ls
abcl Documents equipment memos Music Public stack.yaml Videos
Desktop Downloads letters misk Pictures ski.plases Templates work
[asmaslova@fedora ~]$ cat etc/password
cat: etc/password: No such file or directory
[asmaslova@fedora ~]$ cat /etc/password
cat: /etc/password: No such file or directory

```

рис.4 Настройка прав доступа для файлов

4. Я просмотрела содержимое файла /etc/password, скопировала файл ~/feathers в файл ~/file.old, переместите файл ~/file.old в каталог ~/play, скопируйте каталог ~/play в каталог ~/fun, переместите каталог ~/fun в каталог ~/play и назовите его games. Далее я лишила владельца файла ~/feathers права на чтение и

попробовала просмотреть файл ~/feathers командой cat, но у меня ничего не вышло, поскольку права на чтение ограничены. Скопировать у меня не вышло по этой же причине. Далее я вернула владельцу файла ~/feathers право на чтение и лишила владельца каталога ~/play права на выполнение, после чего попыталась перейти в каталог ~/play, но не смогла из-за ограниченности прав. После этого я вернула права на выполнение владельцу (рис.5-6).

```
[asmaslova@fedora ~]$ cd ski.plases/
[asmaslova@fedora ski.plases]$ ls
australia equipment feathers my_os plans play
[asmaslova@fedora ski.plases]$ cd plans/
[asmaslova@fedora plans]$ ls
[asmaslova@fedora plans]$ cd ..
[asmaslova@fedora ski.plases]$ cd play
[asmaslova@fedora play]$ ls
[asmaslova@fedora play]$ cd ..
[asmaslova@fedora ski.plases]$ cd
[asmaslova@fedora ~]$ cp ~/ski.plases/feathers ~/file.ols
[asmaslova@fedora ~]$ cp ~/ski.plases/feathers ~/file.old
[asmaslova@fedora ~]$ ls
abc1 Documents equipment file.ols memos Music Public stack.yaml Videos
Desktop Downloads file.old letters misk Pictures ski.plases Templates work
[asmaslova@fedora ~]$ rm file.ols
[asmaslova@fedora ~]$ mv ~/file.old ~/ski.plases/play
[asmaslova@fedora ~]$ ls
abc1 Documents equipment memos Music Public stack.yaml Videos
Desktop Downloads letters misk Pictures ski.plases Templates work
[asmaslova@fedora ~]$ cd ski.plases/pla
plans/ play/
[asmaslova@fedora ~]$ cd ski.plases/play
[asmaslova@fedora play]$ ls
file.old
[asmaslova@fedora play]$ cd ..
[asmaslova@fedora ski.plases]$ cd ..
[asmaslova@fedora ~]$ cp ~/ski.plases/play ~/fun
cp: -r not specified; omitting directory '/home/asmaslova/ski.plases/play'
[asmaslova@fedora ~]$ cp -r ~/ski.plases/play ~/fun
[asmaslova@fedora ~]$ ls
abc1 Documents equipment letters misk Pictures ski.plases Templates work
Desktop Downloads fun memos Music Public stack.yaml Videos
[asmaslova@fedora ~]$ cd fun
[asmaslova@fedora fun]$ ls
file.old
[asmaslova@fedora fun]$ cd
[asmaslova@fedora ~]$ mvdir ~/fun ~/ski.plases/play
bash: mvdir: command not found...
Similar command is: 'vdir'
[asmaslova@fedora ~]$ mv ~/fun ~/ski.plases/play
[asmaslova@fedora ~]$ cd ski.plases/play
[asmaslova@fedora play]$ ls
file.old fun
```

рис.5 Выполнение пунктов 4.1-4.5 задания

```

[asmaslova@fedora play]$ ls
file.old  fun
[asmaslova@fedora play]$ mv fun games
[asmaslova@fedora play]$ ls
file.old  games
[asmaslova@fedora play]$ cd ..
[asmaslova@fedora ski.plases]$ ls
australia equipment feathers my_os plans play
[asmaslova@fedora ski.plases]$ chmod u-r feathers
[asmaslova@fedora ski.plases]$ ls -l feathers
--w-rw-r--. 1 asmaslova asmaslova 0 May  7 16:15 feathers
[asmaslova@fedora ski.plases]$ cat feathers
cat: feathers: Permission denied
[asmaslova@fedora ski.plases]$ cp feathers ~/ski.plases/play
cp: cannot open 'feathers' for reading: Permission denied
[asmaslova@fedora ski.plases]$ chmod u+r feathers
[asmaslova@fedora ski.plases]$ ls -l feathers
-rw-rw-r--. 1 asmaslova asmaslova 0 May  7 16:15 feathers
[asmaslova@fedora ski.plases]$ ls
australia equipment feathers my_os plans play
[asmaslova@fedora ski.plases]$ chmod u-x play
[asmaslova@fedora ski.plases]$ cd play
bash: cd: play: Permission denied
[asmaslova@fedora ski.plases]$ ls -l play
ls: cannot access 'play/file.old': Permission denied
ls: cannot access 'play/games': Permission denied
total 0
-????????? ? ? ? ?      ? file.old
d????????? ? ? ? ?      ? games
[asmaslova@fedora ski.plases]$ chmod u+x play
[asmaslova@fedora ski.plases]$ ls -l play
total 0
-rw-rw-r--. 1 asmaslova asmaslova  0 May  7 16:29 file.old
drwx--x--x. 1 asmaslova asmaslova 16 May  7 16:31 games
[asmaslova@fedora ski.plases]$ ls -l
total 0
drwxr--r--. 1 asmaslova asmaslova  0 May  7 16:20 australia
drwxrwxr-x. 1 asmaslova asmaslova 38 May  7 16:13 equipment
-rw-rw-r--. 1 asmaslova asmaslova  0 May  7 16:15 feathers
-r-xr--r--. 1 asmaslova asmaslova  0 May  7 16:15 my_os
drwxrwxr-x. 1 asmaslova asmaslova  0 May  7 16:14 plans
drwx--x--x. 1 asmaslova asmaslova 26 May  7 16:32 play
[asmaslova@fedora ski.plases]$ cd
[asmaslova@fedora ~]$ man mount
[asmaslova@fedora ~]$ man fsck
[asmaslova@fedora ~]$ man mkfs
[asmaslova@fedora ~]$ man kill
[asmaslova@fedora ~]$

```

рис.6 Выполнение пунктов 4.6-4.12 задания

5. Далее я прочитала ман по командам mount, fsck, mkfs, kill и кратко их охарактеризовала, приведя примеры (рис.6-10)

```
mount(8) System Administration MOUNT(8)
NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-hj-v]
    mount [-l] [-t fstype]
    mount -a [-fforward] [-t fstype] [-O optlist]
    mount [-forw] [-o options] device[mountpoint]
    mount [-forw] [-t fstype] [-o options] device mountpoint
    mount --bind|--rbind|--move alldir newdir
    mount --make[shared|slave|private|unbindable|shared|rslave|rprivate|runbindable] mountpoint

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 other 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 is of type type) at the directory dir. The option -t type is optional. The mount command is usually able to detect a filesystem. The root permissions are necessary to mount a filesystem by default. See section "Non-superuser mounts" below for more details. The previous contents (if any) and owner and mode of dir become invisible, and as long as this filesystem remains mounted, the pathname dir refers to the root of the filesystem on device.

    If only the directory or the device is given, for example:

        mount /dir

    then mount looks for a mountpoint (and if not found then for a device) in the /etc/fstab file. It's possible to use the --target or --source options to avoid ambiguous interpretation of the given argument. For example:

        mount --target /mountpoint

    The same filesystem may be mounted more than once, and in some cases (e.g., network filesystems) the same filesystem may be mounted on the same mountpoint multiple times. The mount command does not implement any policy to control this behavior. All behavior is controlled by the kernel and it is usually specific to the filesystem driver. The exception is --all, in this case already mounted filesystems are ignored (see --all below for more details).

    Listing the mounts
    The listing mode is maintained for backward compatibility only.

Manual page mount(8) line 3 (press h for help or q to quit)
```

рис.7 Мануал команды mount

```
fsck(8) System Administration FSCK(8)
NAME
    fsck - check and repair a Linux filesystem

SYNOPSIS
    fsck [-lsavrtmp] [-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=886ab0f6-88c5-4a83-98b8-bfc240577bd 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 status 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
    8      Operational error
    16     Usage or syntax error
    32     Checking canceled by user request
    128    Shared-library error

    The exit status returned when multiple filesystems are checked is the bit-wise OR of the exit statuses for each filesystem that is checked.

    In actuality, fsck is simply a front-end for the various filesystem checkers (fsck.fstype) available under Linux. The filesystem-specific checker is searched for in the PATH environment variable. If the PATH is undefined then fallback to /sbin.

    Please see the filesystem-specific checker manual pages for further details.

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

рис.8 Мануал команды fsck

```
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 status 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
Specify the type of filesystem to be built. If not specified, the default filesystem type (currently ext2) is used.
fs-options
Filesystem-specific options to be passed to the real filesystem builder.
-V, --verbose
Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.
-v, --version
Display version information and exit. (Option -V will display version information only when it is the only parameter, otherwise it will work as --verbose.)
-h, --help
Display help text and exit.
BUGS
All generic options must precede and not be combined with filesystem-specific options. Some filesystem-specific programs do not automatically detect the device size and require the size parameter to be specified.
AUTHORS
David Jongs <davidjongs.com>, Fred N. van Kempen <waltje@u.walt.nl.mugnet.org>, Ron Sommeling <sommel@sci.kun.nl>.
The manual page was shamelessly adapted from Remy Card's version for the ext2 filesystem.
SEE ALSO
fs(5), bsdlocks(8), fsck(8), mkdosfs(8), mke2fs(8), mkfs.bfs(8), mkfs.ext2(8), mkfs.ext3(8), mkfs.ext4(8), mkfs.minix(8), mkfs.msdos(8), mkfs.vfat(8), mkfs.xfs(8)
REPORTING BUGS
Manual page mkfs(8) line 1 (press h for help or q to quit)
```

рис.9 Мануал команды *mkfs*

```
kill(1) User Commands kill(1)
NAME
kill - terminate a process
SYNOPSIS
kill [-signal|--signal=-p] [-q value] [-w] [--timeout milliseconds signal] [--] pid|name...
kill -l [number] | -L
DESCRIPTION
The command kill sends the specified signal to the specified processes or process groups.
If no signal is specified, the TERM signal is sent. The default action for this signal is to terminate the process. This signal should be used in preference to the KILL signal (number 9), since a process may install a handler for the TERM signal in order to perform clean-up steps before terminating in an orderly fashion. If a process does not terminate after a TERM signal has been sent, then the KILL signal may be used; be aware that the latter signal cannot be caught, and so does not give the target process the opportunity to perform any clean-up before terminating.
Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.
If signal is 0, then no actual signal is sent, but error checking is still performed.
ARGUMENTS
The list of processes to be signaled can be a mixture of names and PIDs.
pid
Each pid can be expressed in one of the following ways:
0
where n is larger than 0. The process with PID n is signaled.
0
All processes in the current process group are signaled.
-1
All processes with a PID larger than 1 are signaled.
-n
where n is larger than 1. All processes in process group n are signaled. When an argument of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '-' option, otherwise it will be taken as the signal to send.
name
All processes invoked using this name will be signaled.
OPTIONS
-s, --signal signal
The signal to send. It may be given as a name or a number.
Manual page kill(1) line 1 (press h for help or q to quit)
```

рис.10 Мануал команды *kill*

Выводы

По итогам лабораторной работы я ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов, а также на практике научилась применять команды для работы с файлами и каталогами, управлять процессами (и работами), проверять использование диска и обслуживать файловую систему.

Список литературы