

Лабораторная работа №7

Анализ файловой системы Linux. Команды для работы с файлами и каталогами

Буллер Т. А.

23 марта 2024

Российский университет дружбы народов, Москва, Россия

Информация

- Буллер Татьяна Александровна
- студент группы НБИбд-01-23
- Российский университет дружбы народов

Вводная часть

- Файловая система Linux
- команды для работы с файлами и каталогами
- команды по управлению процессами (и работами)
- команды по проверке использования диска и обслуживанию файловой системы

- 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.

- Виртуальная машина Kali Linux
- Терминал bash
- Процессор pandoc для входного формата Markdown
- Результирующие форматы
 - pdf
 - html
- Автоматизация процесса создания: Makefile

Выполнение заданий примеров

Необходимо скопировать файл `~/abc1` в файл `april` и в файл `may`. Для этого командой `touch` создаем сам файл, далее с помощью команды `cp` копируем его содержимое в новые файлы.

Копирование и создание файлов.

```
(tabuller@jordi)-[~]  
$ touch abc1
```

```
(tabuller@jordi)-[~]  
$ cp abc1 may
```

```
(tabuller@jordi)-[~]  
$ cp abc1 april
```

```
(tabuller@jordi)-[~]  
$ mkdir monthly
```

Теперь нужно скопировать файлы `april` и `may` в каталог `monthly` - для этого с помощью `mkdir` создаем каталог `monthly`, с помощью уже известной `cp` копируем файлы в каталог. Припишем названия файлов в одну строку - так они скопируются в рамках одной команды.

```
(tabuller@jordi)-[~]  
$ mkdir monthly
```

```
(tabuller@jordi)-[~]  
$ cd april may monthly  
cd: too many arguments
```

```
(tabuller@jordi)-[~]  
$ cp april may monthly
```

После этого скопируем файл `monthly/may` в файл с именем `june` - для этого используем команду `cp` и прописываем полный путь до файла.

```
(tabuller@jordi)-[~]  
$ cp april may monthly  
  
(tabuller@jordi)-[~]  
$ cp monthly/may monthly/june  
  
(tabuller@jordi)-[~]  
$ ls monthly  
april  june  may
```

Рис. 3: Копирование файлов в произвольном каталоге

Теперь скопируем каталог `monthly` в каталог `monthly.00`: для этого используем `cp -r`. Опция копируем рекурсивно - весь каталог перемещается в созданный новый.

```
(tabuller@jordi)-[~]  
$ mkdir monthly.00  
  
(tabuller@jordi)-[~]  
$ cp -r monthly monthly.00  
  
(tabuller@jordi)-[~]  
$ ls monthly.00  
monthly
```

Рис. 4: Копирование каталогов в текущем каталоге

Далее по заданию нужно скопировать каталог `monthly.00` в каталог `/tmp`. Для этого опять используем опцию рекурсивного копирования и прописываем полный путь к файлу (каталогу), куда хотим копировать.

Копирование и создание файлов.

```
(tabuller@jordi)-[~]  
$ cp -r monthly.00 /tmp
```

```
(tabuller@jordi)-[~]  
$ ls
```

```
abcl      Desktop      Downloads    LIC  
april     Documents    image        may
```

```
(tabuller@jordi)-[~]  
$ ls /tmp
```

```
monthly.00
```

Перемещение и переименование файлов и каталогов

Задание: зменить название файла `april` на `july` в домашнем каталоге. Для этого используем команду `mv`: де-юре название будет изменено, де-факто мы перемещаем файл в файл с другим названием и удаляем предыдущий одной командой.

Перемещение и переименование файлов и каталогов

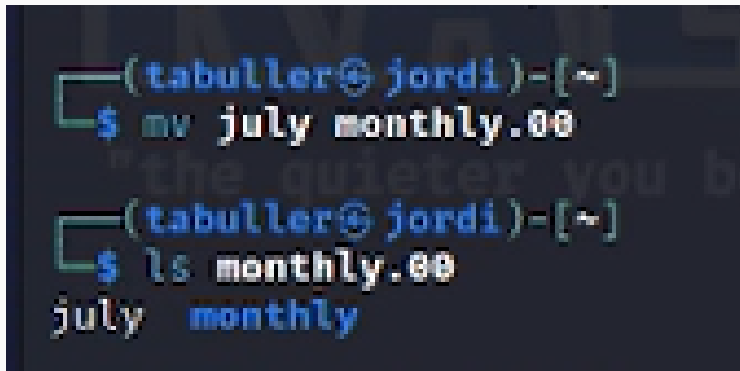
```
(tabuller@jordi)-[~]  
$ mv april july  
  
(tabuller@jordi)-[~]  
$ ls  
abc1    Documents image LICENSE monthly Music Public ssh1.pub Videos  
Desktop Downloads july    may    monthly.00 Pictures ssh1 Templates work
```

Рис. 6: Переименование файлов в текущем каталоге

Перемещение и переименование файлов и каталогов

Чтобы переместить файл `july` в каталог `monthly.00` используем ту же команду, на этот раз прописывая полный путь до места назначения. При проверке новый файл появляется вместе с остальным содержимым каталога.

Перемещение и переименование файлов и каталогов

A terminal window with a dark background and light blue text. The prompt is (tabuller@jordi)-[~]. The first command is \$ mv july monthly.00. The second prompt is (tabuller@jordi)-[~]. The second command is \$ ls monthly.00. The output is july monthly.

```
(tabuller@jordi)-[~]  
$ mv july monthly.00  
  
(tabuller@jordi)-[~]  
$ ls monthly.00  
july  monthly
```

Рис. 7: Перемещение файлов в другой каталог

Перемещение и переименование файлов и каталогов

Необходимо также переименовать каталог `monthly.00` в `monthly.01`. Для этого снова используем `mv`, работаем точно так же, как когда нам нужно было переименовать файл.

Перемещение и переименование файлов и каталогов

```
(tabuller@jordi)-[~]  
$ mv monthly.00 monthly.01  
  
(tabuller@jordi)-[~] u become, the more you are able to hear"  
$ ls  
abc1      Documents image    may      monthly.01 Pictures  ssh1      Templates work  
Desktop  Downloads LICENSE  monthly Music     Public    ssh1.pub  Videos
```

Рис. 8: Переименование каталогов в текущем каталоге

Перемещение и переименование файлов и каталогов

Переместить каталог `monthly.01` в каталог `reports`. Для этого сперва создаем каталог с нужным именем, после этого перемещаем каталог, как до этого делали с файлами. Чтобы далее переименовать каталог `reports/monthly.01` в `reports/monthly` снова используем ту же команду, но на этот раз, так как мы переименовываем файлы не в рабочем каталоге, прописываем полный путь до них:

Перемещение и переименование файлов и каталогов

```
(tabuller@jordi)-[~]  
$ mkdir reports | mv monthly.01 reports  
  
(tabuller@jordi)-[~]  
$ ls reports  
monthly.01  
  
(tabuller@jordi)-[~]  
$ mv reports/monthly.01 reports/monthly  
  
(tabuller@jordi)-[~]  
$ ls reports  
monthly
```

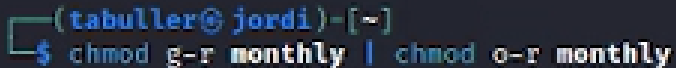
Рис. 9: Перемещение каталога в другой каталог. Переименование каталога, не

В каталоге существует файл `may`. Дадим владельцу права на его исполнение, после чего заберем назад и посмотрим, что получилось. После добавления права исполнения (`+u`) для владельца (`u`) при просмотре файла он подсвечивается зеленым - теперь его можно исполнять, а в строке прав появляется дополнительный символ. Когда же право изымается, файл возвращается к исходному состоянию.

```
(tabuller@jordi)-[~]  
$ ls -l may  
-rw-r--r-- 1 tabuller buller 0 Mar 23 03:06 may  
  
(tabuller@jordi)-[~]  
$ chmod u+x may  
  
(tabuller@jordi)-[~]  
$ ls -l may  
-rwxr--r-- 1 tabuller buller 0 Mar 23 03:06 may  
  
(tabuller@jordi)-[~]  
$ chmod u-x may  
  
(tabuller@jordi)-[~]  
$ ls -l may  
-rw-r--r-- 1 tabuller buller 0 Mar 23 03:06 may
```

Рис. 10: Добавление и удаление права исполнения

Требуется создать каталог `monthly` с запретом на чтение для членов группы и всех остальных пользователей. Каталог уже создан, для того, чтобы лишить пользователей прав чтения, используем `chmod`. `-r` - лишение права чтения, для группы (`g`) и всех пользователей (`o`) соответственно.

A terminal window with a dark background. The prompt is `(tabuller@jordi)-[~]`. The command entered is `$ chmod g-r monthly | chmod o-r monthly`.

```
(tabuller@jordi)-[~]  
$ chmod g-r monthly | chmod o-r monthly
```

Рис. 11: Изъятие права чтения для группы и других пользователей

Теперь создадим новый файл и дадим группе права на запись в него.
Пользуемся тем же, чем пользовались, когда добавляли владельцу права на исполнение, но теперь работаем с группой (g) и правом записи (+w)

```
(tabuller@jordi)-[~]  
$ touch abc1  
  
(tabuller@jordi)-[~]  
$ chmod g+w abc1  
  
(tabuller@jordi)-[~]  
$ ls -l abc1  
-rw-rw-r-- 1 tabuller buller 0 Mar 23 03:17 abc1
```

Рис. 12: Добавление права записи для группы

Выполнение лабораторной работы

Задание: скопируйте файл (...) в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него. Проблема: моя система не имеет каталога `/usr/include/sys/` в принципе. Вместо этого копируем файл из `usr/include/`, проверяем его наличие - в выводе команды `ls` он отображается - и переходим к следующему шагу: создадим новую директорию и переместим скопированный файл в него.

Копирование и создание файлов.

```
(tabuller@jordi)-[~]  
$ cp /usr/include/aio.h ~/equipment  
  
(tabuller@jordi)-[~]  
$ ls  
abc1    Documents  equipment  LICENSE  monthly  Pictures  reports  ssh1.pub  Videos  
Desktop Downloads  image     may      Music    Public    ssh1     Templates work  
  
(tabuller@jordi)-[~]  
$ mkdir ski.places  
  
(tabuller@jordi)-[~]  
$ ls  
abc1    Downloads  LICENSE  Music    reports  ssh1.pub  work  
Desktop equipment  may      Pictures  ski.places  Templates  
Documents image     monthly  Public    ssh1       Videos  
  
(tabuller@jordi)-[~]  
$ mv equipment ski.places/equipment
```

Рис. 13: Копирование и перемещение файла

Далее скопируем под другим названием файл `abc1`, для этого используем команду `cp`. Проверим - файлы скопировались, все на месте. Далее создаем новую директорию и переносим новые файлы туда.

Копирование и создание файлов.

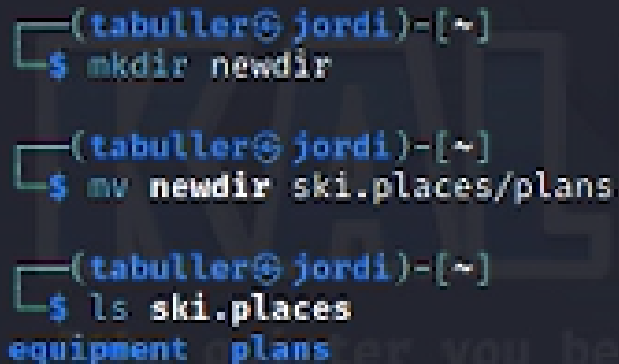
```
(tabuller@jordi)-[~]  
$ cp abc1 ski.places/equiplist2  
  
(tabuller@jordi)-[~]  
$ ls ski.places  
equiplist  equiplist2  
  
(tabuller@jordi)-[~]  
$ mkdir ski.places/equipment  
  
(tabuller@jordi)-[~]  
$ ls ski.places  
equiplist  equiplist2  equipment  
  
(tabuller@jordi)-[~]  
$ mv ski.places/equiplist ski.places/equiplist2 ski.places/equipment  
  
(tabuller@jordi)-[~]  
$ ls ski.places  
equipment
```

Рис. 14: Копирование и перемещение файлов

Перемещение и переименование файлов и каталогов

Теперь создадим в домашнем каталоге новую директорию и переместим ее (уже под другим именем) в созданный каталог `ski.places` - для этого используем `mkdir` и `mv` соответственно.

Перемещение и переименование файлов и каталогов

A terminal window with a dark background and light blue text. It shows three commands being executed in sequence. The first command creates a directory named 'newdir'. The second command moves 'newdir' to 'ski.places/plans'. The third command lists the contents of 'ski.places', showing 'equipment' and 'plans'.

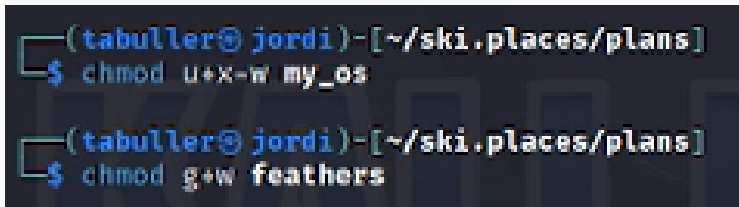
```
(tabuller@jordi)-[~]  
$ mkdir newdir  
  
(tabuller@jordi)-[~]  
$ mv newdir ski.places/plans  
  
(tabuller@jordi)-[~]  
$ ls ski.places  
equipment plans
```

Рис. 15: Создание и перемещение нового каталога

Далее определим опции, необходимые для изменение прав на файлы и директории. Что имеем по заданию: - drwxr--r-- ... australia - drwx--x--x ... play - -r-xr--r-- ... my_os - -rw-rw-r-- ... feathers

Видим, что первые два файла являются директориями, последние - отдельными файлами. Начнем с конца. Изначально владелец файла имеет право на чтение и запись, все остальные (группа и прочие пользователи) - только на чтение. Для файлов нам необходимо:

- убрать у владельца `my_os` право на запись, добавить право на исполнение (`u-w+x`)
- добавить группе владельца файла `feathers` право на запись в файл (`g+w`)



```
(tabuller@jordi)-[~/ski.places/plans]  
$ chmod u+x-w my_os  
  
(tabuller@jordi)-[~/ski.places/plans]  
$ chmod g+w feathers
```

Рис. 16: Редактирование доступа к файлам

australia и play являются каталогами, о чем говорит буква d в начале строки прав доступа. По умолчанию владелец директории имеет все права к ней, группа и другие пользователи - права на чтение и исполнение.

Для первого каталога необходимо убрать у всех, кроме владельца, права на исполнение. Для этого используем `go-x`. Для второго каталога убираем у всех, кроме владельца, права на чтение `go-w`.

```
(tabuller@jordi)-[~/ski.places/plans]
$ mkdir australia play

(tabuller@jordi)-[~/ski.places/plans]
$ chmod u+x australia

(tabuller@jordi)-[~/ski.places/plans]
$ chmod u+x play | chmod go+x-r play

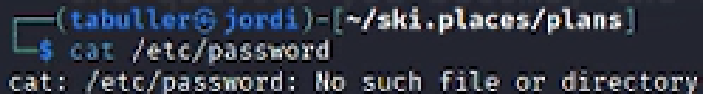
(tabuller@jordi)-[~/ski.places/plans]
$ ls -l
total 8
drwxr-xr-x 2 tabuller buller 4096 Mar 23 03:29 australia
-rw-rw-r-- 1 tabuller buller    0 Mar 23 03:23 feathers
-r-xr--r-- 1 tabuller buller    0 Mar 23 03:23 my_os
drwx--x--x 2 tabuller buller 4096 Mar 23 03:29 play

(tabuller@jordi)-[~/ski.places/plans]
$ chmod go-x australia
```

Рис. 17: Редактирование доступа к каталогам

Теперь продолжим работу с файлами. Попробуем вывести на экран содержимое файла `/etc/password` - и столкнемся с ошибкой, потому что на данном дистрибутиве такого файла в системе нет. Его функции выполняет `/etc/passwd`.

Перемещение и переименование файлов и каталогов

A terminal window with a dark background. The prompt is `(tabuller@jordi)-[~/ski.places/plans]`. The user enters `$ cat /etc/password`. The output is `cat: /etc/password: No such file or directory`.

```
(tabuller@jordi)-[~/ski.places/plans]  
$ cat /etc/password  
cat: /etc/password: No such file or directory
```

Рис. 18: password, не passwd

Перемещение и переименование файлов и каталогов

Далее скопируем уже имеющийся файл `feathers` под новым названием в домашнюю директорию. `file.old` теперь перемещаем в директорию `play`, копируем этот каталог в созданный в домашнем каталоге `fun` и уже его переносим назад в `play/games`

Перемещение и переименование файлов и каталогов

```
(tabuller@jordi)-[~]  
$ cp ski.places/plans/feathers ~/file.old  
  
(tabuller@jordi)-[~]  
$ mv file.old ski.places/plans/play  
  
(tabuller@jordi)-[~]  
$ cp ski.places/plans/play ~/fun  
cp: -r not specified; omitting directory 'ski.places/plans/play'  
  
(tabuller@jordi)-[~]  
$ cp -r ski.places/plans/play ~/fun  
  
(tabuller@jordi)-[~]  
$ mv fun ski.places/plans/play/games
```

Рис. 19: Махинации с копированием и перемещением файлов и каталогов

Теперь поработаем с правами доступа. Лишим владельца файла ~/feathers права на чтение. При просмотреть файл ~/feathers командой cat мы встречаемся с ошибкой: доступ запрещен, мы только что отобрали у себя это право. То же самое будет при попытке скопировать файл ~/feathers. Смилуемся и вернем право на чтение: u+r.

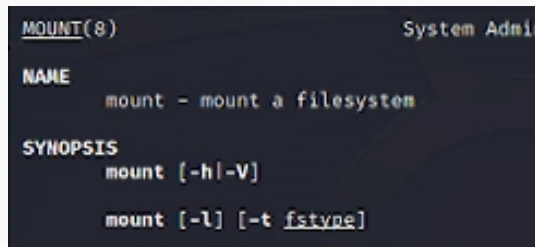
Далее лишаем владельца каталога ~/play права на выполнение. При попытке перейти в этот каталог снова видим ту же ошибку: доступ запрещен, так как я запретила владельцу (себе) исполнять этот файл (каталог). Возвращаю права на выполнение - теперь все будет работать.

Права доступа и их изменение

```
(tabuller@jordi)-[~]  
$ chmod u-r ski.places/plans/feathers  
  
(tabuller@jordi)-[~]  
$ cat ski.places/plans/feathers  
cat: ski.places/plans/feathers: Permission denied  
  
(tabuller@jordi)-[~]  
$ chmod u+x ski.places/plans/feathers  
  
(tabuller@jordi)-[~]  
$ cp ski.places/plans/feathers feathers  
cp: cannot open 'ski.places/plans/feathers' for reading: Permission denied  
  
(tabuller@jordi)-[~]  
$ chmod u+r ski.places/plans/feathers | chmod u-x ski.places/plans/play  
  
(tabuller@jordi)-[~]  
$ cd ski.places/plans/play  
cd: permission denied: ski.places/plans/play more you are able to  
  
(tabuller@jordi)-[~]  
$ chmod u+x ski.places/plans/play
```

Рис. 20: Махинации с правами доступа файлов и каталогов

После этого посмотрим на несколько руководств. Команда `mount`, судя по описанию, монтирует файловую систему:

A screenshot of a terminal window showing the man page for the 'mount' command. The title is 'MOUNT(8)' and the user is 'System Admin'. The 'NAME' section states 'mount - mount a filesystem'. The 'SYNOPSIS' section shows two command formats: 'mount [-h|-V]' and 'mount [-l] [-t fstype]'.

```
MOUNT(8)                                     System Admin

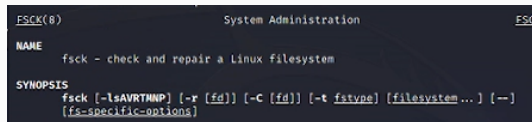
NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]
```

Рис. 21: руководство к команде `mount`

Команда `fsck` проверяет файловую систему и репарировает ее, если были найдены ошибки.

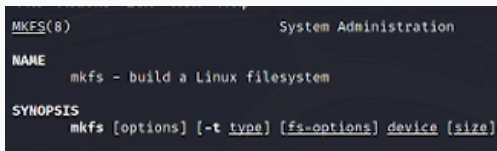
A screenshot of a terminal window displaying the manual page for the 'fsck' command. The title bar shows 'FSCK(8)' on the left, 'System Administration' in the center, and 'FSCK' on the right. The content is as follows:

```
NAME
  fsck - check and repair a Linux filesystem

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

Рис. 22: руководство к команде `fsck`

Команда mkfs создает файловую систему в принципе.

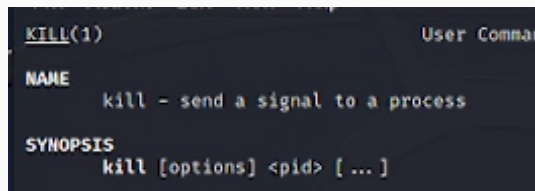


```
mkfs(8)                                System Administration
NAME
    mkfs - build a Linux filesystem

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

Рис. 23: руководство к команде mkfs

Команда kill “убивает” (отключает) процессы или задачи.

A screenshot of a terminal window showing the man page for the 'kill' command. The text is as follows:

```
KILL(1) User Commands  
  
NAME  
    kill - send a signal to a process  
  
SYNOPSIS  
    kill [options] <pid> [ ... ]
```

Рис. 24: руководство к команде kill

Выводы

Приобретены практические навыки взаимодействия пользователя с системой посредством командной строки.