

Задание 1: Создать файл file1 и наполнить его произвольным содержимым. Скопировать его в file2. Создать символическую ссылку file3 на file1. Создать жесткую ссылку file4 на file1. Посмотреть, какие айноды у файлов. Удалить file1. Что стало с остальными созданными файлами? Попробовать вывести их на экран.

Для создания и наполнения файла воспользуюсь cat

```
cat > file1
#набор текста
#многострочный текст
#еще строки текста
нажатие Ctrl+D завершает ввод текста
```

Копирование, создание жёсткой и символической ссылок

```
cp file1 file2 #копирование файла
ln -s file1 file3 #символическая ссылка
ln file1 file4 #жесткая ссылка
ls -l #вывод списка файлов для просмотра inode
```

При данных командах inode файлов file1, file2, file3 различаются. inode file1 и file4 совпадают.

При удалении file1 имеем следующую ситуацию:

file2 - был скопирован, открывается, выводится на экран терминала.

file3 - cat: file3: Нет такого файла или каталога (символическая ссылка на отсутствующий файл)

file4 - имеет inode аналогичную file1, открывается, выводится на экран терминала.

Задание 2: Дать созданным файлам другие, произвольные имена. Создать новую символическую ссылку. Переместить ссылки в другую директорию.

Для начала вернул file1 -> \$ln file4 file1 - таким образом символическая ссылка file3 будет возвращена.

```
mv file1 testfile
mv file2 movedfile
mv file3 moved_symbol
mv file4 samename #переименование файлов
```

Переименование file1 сносит символическую ссылку file3. Переименованные file1 и file4 сохраняют равные inode.

ln -s '/home/meshd/testfile' 'testfolder/1/2/movedfile2' #перенос файла с символической ссылкой.

ln '/home/meshd/testfile' 'testfolder/1/2/movedfile3' #перенос файла с жесткой ссылкой

Создание символической ссылки на файл требует абсолютного пути на файл-объект для переноса и относительный путь к файлу-результату. Для переноса файла с жесткой ссылкой будет достаточно указать имя файла объекта переноса и относительный путь к файлу-результату. При создании жёсткой ссылки в другой директории inode файла результат так же равен файлу-объекту.

Задание 3: Создать два произвольных файла. Первому присвоить права на чтение, запись для владельца и группы, только на чтение — для всех. Второму присвоить права на чтение, запись — только для владельца. Сделать это в численном и символьном виде.

Для выполнения задания создаются директория и файлы в домашней директории.

```
mkdir task3
cd task3
cat > task3_1
#произвольный текст
ctrl+D
cat > task3_2
#произвольный текст
ctrl+D
ll #проверка прав доступа и привязки файлов к группе и пользователям
```

На момент создания файлы относятся к пользователю meshd и группе meshd, имеют права доступа rw- rw- r-- :

Пользователь meshd

Группа meshd

Для выполнения задания task3_1 должен иметь права доступа rw- rw- r--
task3_2 должен иметь права доступа rw- --- ---

```
chmod -rwx [file] #снятие прав доступа с файла
chmod u=rw,g=rw,o=r task3_1 #назначение прав доступа к файлу
chmod 664 task3_1 #команда аналогична команде выше
chmod u+rw task3_2 #назначение прав доступа к файлу
chmod 600 task3_2 #команда аналогична команде выше
```

Задание 4: Создать пользователя, обладающего возможностью выполнять действия от имени суперпользователя.

Для выполнения данной задачи воспользуюсь данной командой:

```
sudo useradd -m -G task4user -s /bin/bash task4user  
sudo usermod -a -G sudo task4user
```

Данная команда создаст пользователя task4user, далее команда добавит пользователя в группу sudo с соответствующими правами.

Задание 5*: Создать группу developer и несколько пользователей, входящих в нее. Создать директорию для совместной работы. Сделать так, чтобы созданные одними пользователями файлы могли изменять другие пользователи этой группы.

```
groupadd developer #создание группы developer  
sudo useradd -m -G developer -s /bin/bash ivanov #создание пользователя ivanov  
sudo passwd #создание пароля для пользователя  
sudo useradd -m -G developer -s /bin/bash petrov #создание пользователя petrov  
sudo passwd #создание пароля для пользователя  
mkdir /home/developer #создание группы для новых пользователей  
chown -R :developer /home/developer #присвоение директории для группы developer  
chmod 770 /home/developer #присвоение правил прав доступа к директории developer
```

Данный ряд команд позволяет дать доступ к директории только пользователям группы developer.

Примечание: не смог через useradd и ключи -d -b создать домашний и базовый директории, пробовал через них изначально создать общую домашнюю и общую директорию для пользователей группы. Если возможно, то можно будет этот момент рассмотреть на разборе ДЗ на занятии (-b и -d флаги).

Задание 6*: Создать в директории для совместной работы поддиректорию для обмена файлами, но чтобы удалять файлы могли только их создатели.

Для данного задания воспользуюсь результатами предыдущего задания 5*. В директории /home/developer

cd developer #переход для дальнейшего действия в директории по
относительному пути
mkdir -p -m 775 sharing #создание директории для обмена файлами

Проверка созданием файла от пользователя ivanov и дальнейшего перехода на пользователя retrov с попыткой удаления файла через команду rm "file" выдает ошибку отказа в доступе.

**Задание 7*: Создать директорию, в которой есть несколько файлов.
Сделать так, чтобы открыть файлы можно было, только зная имя файла, а через
ls список файлов посмотреть было нельзя.**

Для данного задания воспользуюсь результатами предыдущего задания 5*. В директории /home/developer.

cd /home/developer #переход для дальнейшего действия в
директории по абсолютному пути
mkdir task7 #создание директории для задания 7
cat > task7/text_task7 #создание и заполнение файла для задания
chmod 676 task7 #заккрытие для доступа к директории task7

Примечание: Понимаю, что данное решение не является верным для задания 7*.
Другой информации для скрытия директории и не возможности её отображения через ls не нашёл.

В данном варианте решения задания нет возможности получить доступ к содержимому директории task7 через простое "cd task7". Доступ к содержимому файлов находящихся в директории task7 возможно только с указанием пути и префиксом sudo.