

Домашнее задание:

1. Создать файл file1 и наполнить его произвольным содержимым. Скопировать его в file2. Создать символическую ссылку file3 на file1. Создать жёсткую ссылку file4 на file1. Посмотреть, какие inode у файлов. Удалить file1. Что стало с остальными созданными файлами? Попробовать вывести их на экран.
2. Дать созданным файлам другие, произвольные имена. Создать новую символическую ссылку. Переместить ссылки в другую директорию.
3. Создать два произвольных файла. Первому присвоить права на чтение и запись для владельца и группы, только на чтение — для всех. Второму присвоить права на чтение и запись только для владельца. Сделать это в численном и символьном виде.
4. * Создать группу developer и нескольких пользователей, входящих в неё. Создать директорию для совместной работы. Сделать так, чтобы созданные одними пользователями файлы могли изменять другие пользователи этой группы.
5. * Создать в директории для совместной работы поддиректорию для обмена файлами, но чтобы удалять файлы могли только их создатели.
6. * 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.

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

Создаем файл (echo 'Это тестовый файл, созданный Echo' > file1.txt). Сразу, запишем в него фразу.

Скопируем его в «file2.txt» (cp file1.txt file2.txt). Посмотрим inode по обоим файлам (ls -li file1.txt) – «409570» и (ls -li file2.txt) – «409571»

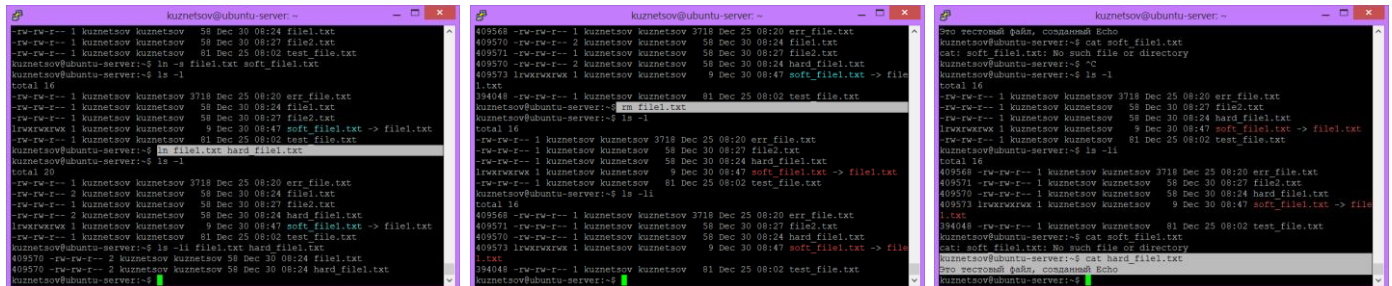
Создадим символическую (софтверную) ссылку на «file1.txt» командой (ln -s file1.txt soft_file1.txt). Убедимся, что тип нового файла («l») и он указывает на «file1.txt». Он имеет отдельный inode. Его можно редактировать...

```
kuznetsov@ubuntu-server: ~  
ls -lx-xr-x 113 root root 0 Dec 30 07:07 proc  
drwxr-xr-x 7 root root 4096 Dec 25 10:34 root  
ls -lxr-xr-x 28 root root 920 Dec 30 07:56 run  
ls -lxr-xr-x 1 root root 8 Jul 31 16:28 shim -> usr/shim  
ls -lxr-xr-x 6 root root 4096 Dec 15 13:17 snap  
ls -lxr-xr-x 2 root root 4096 Jul 31 16:28 srv  
ls -lxr-xr-x 1 root root 178782000 Dec 15 11:59 snap.img  
ls -lx-xr-x 13 root root 0 Dec 30 07:07 sys  
ls -lxr-xr-x 11 root root 4096 Dec 30 07:56 var  
ls -lxr-xr-x 14 root root 4096 Jul 31 16:28 var  
ls -lxr-xr-x 13 root root 4096 Jul 31 16:30 var  
kuznetsov@ubuntu-server:~$ cd ~  
kuznetsov@ubuntu-server:~$ ls -li  
total 9  
-rw-rw-r-- 1 kuznetsov kuznetsov 3718 Dec 25 08:20 err_file.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 81 Dec 25 08:02 test_file.txt  
kuznetsov@ubuntu-server:~$ echo 'Это тестовый файл, созданный Echo' > file1.txt  
kuznetsov@ubuntu-server:~$ ls -li  
total 12  
-rw-rw-r-- 1 kuznetsov kuznetsov 3718 Dec 25 08:20 err_file.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:24 file1.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 81 Dec 25 08:02 test_file.txt  
kuznetsov@ubuntu-server:~$ cp file1.txt file2.txt  
kuznetsov@ubuntu-server:~$ ls -li  
total 16  
-rw-rw-r-- 1 kuznetsov kuznetsov 3718 Dec 25 08:20 err_file.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:24 file1.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:27 file2.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 81 Dec 25 08:02 test_file.txt  
kuznetsov@ubuntu-server:~$ ls -li file1.txt  
409570 -rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:24 file1.txt  
kuznetsov@ubuntu-server:~$ ls -li file2.txt  
409571 -rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:27 file2.txt  
kuznetsov@ubuntu-server:~$ ln -s file1.txt soft_file1.txt  
kuznetsov@ubuntu-server:~$ ls -li  
total 16  
-rw-rw-r-- 1 kuznetsov kuznetsov 3718 Dec 25 08:20 err_file.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:24 file1.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:27 file2.txt  
-rw-rw-r-- 1 kuznetsov kuznetsov 81 Dec 25 08:02 test_file.txt  
kuznetsov@ubuntu-server:~$ ls -li soft_file1.txt  
409572 -rw-rw-r-- 1 kuznetsov kuznetsov 58 Dec 30 08:27 soft_file1.txt -> file1.txt  
kuznetsov@ubuntu-server:~$
```

Создадим жёсткую ссылку на «file1.txt» командой (ln file1.txt hard_file1.txt). Убедимся, что у обоих файлов один inode – «409570» (ls -l file1.txt hard_file1.txt), т.е. нового объекта-файла не создали.

Удаляем файл (rm file1.txt) и посмотрим каталог (ls -l), (ls -li). Естественно, файл «file1.txt» исчез из списка. Символьная (софтверная) ссылка подсвечивает красным (ссылается на несуществующий объект).

При попытке вывести содержимое по софтверной ссылке (cat soft_file1.txt), получим «No such file or directory», что вполне логично. Софтверные ссылки ссылались на имя файла, который удалили. А вот при выводе содержимого жесткой ссылки (cat hard_file1.txt) мы увидим содержимое нами удаленного файла «file1.txt» (жесткая ссылка указывает на inode).



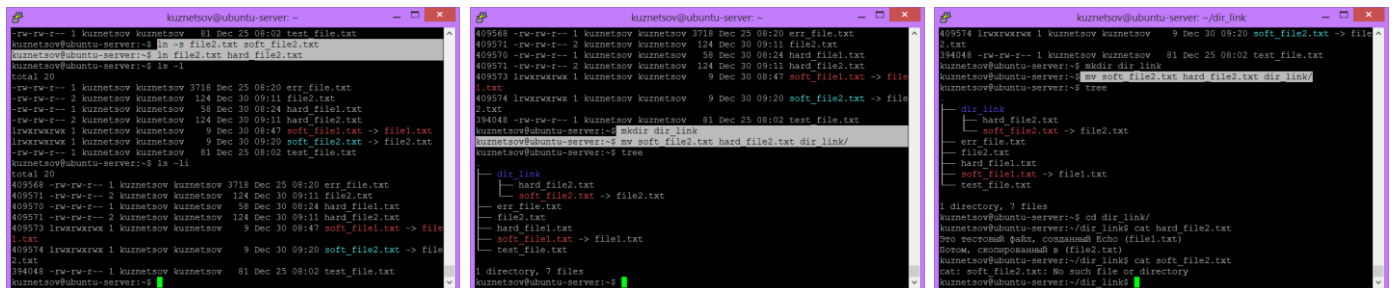
The first terminal window shows the creation of a hard link 'hard_file1.txt' from 'file1.txt' and the removal of 'file1.txt'. The second window shows the output of 'ls -li', highlighting that both files share the same inode (409570). The third window shows the output of 'ls -l', where the soft link 'soft_file1.txt' is highlighted in red, indicating it points to a non-existent file.

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

Создаем софтверную (ln -s file2.txt soft_file2.txt) и жесткую (ln file2.txt hard_file2.txt) ссылки на «file2.txt».

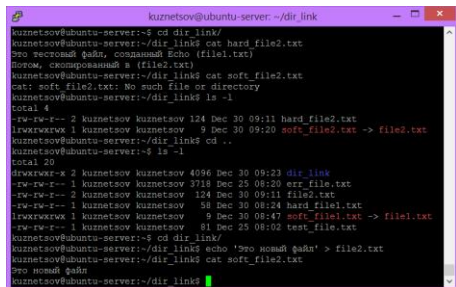
Создаем каталог для ссылок (mkdir dir_link), переместим ссылки в этот каталог (mv soft_file2.txt hard_file2.txt dir_link/).

Теперь, попробуем просмотреть содержимое по ссылкам. По жесткой ссылке – все отлично, содержимое файла выводится. А вот по символической – видим знакомую надпись «No such file or directory».



The first terminal window shows the creation of soft and hard links for 'file2.txt'. The second window shows the creation of a directory 'dir_link' and the movement of the links into it. The third window shows the output of 'ls -li' and 'ls -l' in the 'dir_link' directory, with the soft link highlighted in red.

Проверим догадку и создадим в текущем каталоге др. файл с именем «file2.txt». Снова попробуем посмотреть содержимое софтверной ссылки «soft_file2.txt» - все верно. Теперь эта ссылка снова «рабочая» и ссылается на «новый» файл в текущем каталоге.



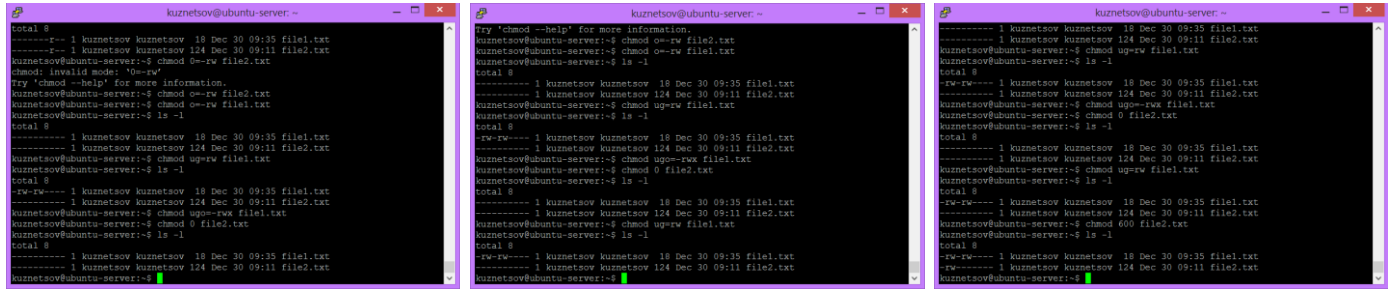
The terminal window shows the creation of a new 'file2.txt' in the current directory and the successful verification of the soft link 'soft_file2.txt' pointing to it.

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

Имеем два файла «file1.txt» и «file2.txt» с «пустыми» правами (chmod ugo=-rwx file1.txt) или (chmod a=-rwx file1.txt) и (chmod 0 file2.txt).

Установим права для «file1.txt» на чтение и запись для владельца и группы командой (chmod ug=rw file1.txt)

Установим права для «file2.txt» на чтение и запись для владельца командой (chmod 600 file2.txt)



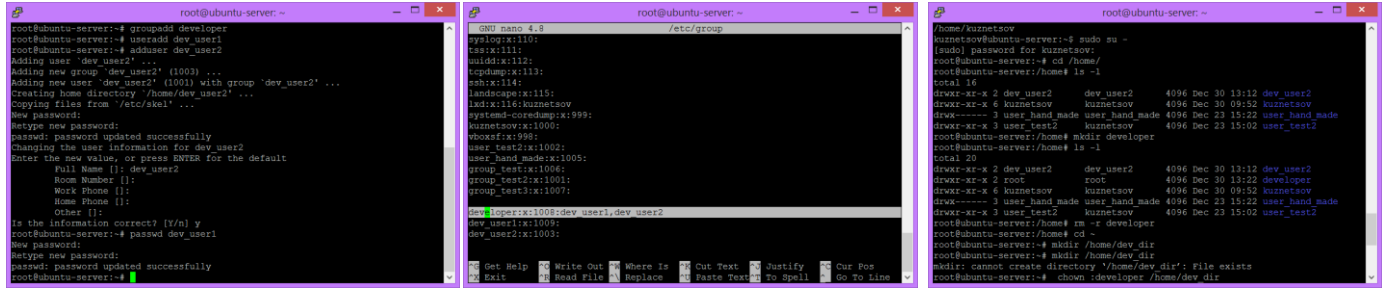
Число	Разрешения	Символьное обозначение
0	разрешения отсутствуют	---
1	x — запуск	--x
2	w — изменение	-w-
3	x+w — запуск+изменение	-wx
4	r — чтение	r--
5	r+x — чтение+запуск	r-x
6	r+w — чтение+изменение	rw-
7	r+w+x — чтение+изменение+запуск	rwx

4. * Создать группу developer и нескольких пользователей, входящих в неё. Создать директорию для совместной работы. Сделать так, чтобы созданные одними пользователями файлы могли изменять другие пользователи этой группы.

Создаем группу (groupadd developer) и пользователей (useradd dev_user1) + (passwd dev_user1) и (adduser dev_user2). У пользователей основные группы «dev_user1» и «dev_user2» соответственно.

Добавим пользователей (usermod -a -G developer dev_user1) и (usermod -a -G developer dev_user2) в группу «developer». Проверим (nano /etc/group) – все Ок.

Создаем каталог для совместной работы «/home/dev_dir» (mkdir /home/dev_dir). Меняем владельца папки (chown :developer /home/dev_dir).



Заходим под «dev_user1», создаем файл «file_user1.txt» (echo 'Этот файл создал dev_user1' > file_user1.txt). Убедимся, что у файла выставилась группа «developer».

```
root@ubuntu-server:~#
mkdir: cannot create directory '/home/dev_dir': file exists
root@ubuntu-server:~# chown developer /home/dev_dir
root@ubuntu-server:~# ls -l /home/dev_dir
total 0
root@ubuntu-server:~# ls /home/dev_dir
root@ubuntu-server:~# tree /home
/home
├── dev_dir
│   ├── dev_user2
│   └── kuznetsov
│       ├── file1.txt
│       ├── file2.txt
│       └── file2.txt.save
├── user_hand_made
└── user_test2

5 directories, 3 files
root@ubuntu-server:~# ls -lha /home/dev_dir
total 0.0K
dwxr-xr-x 2 root developer 4.0K Dec 30 13:25
dwxr-xr-x 2 root developer 4.0K Dec 30 13:25
root@ubuntu-server:~# chmod g+w /home/dev_dir
root@ubuntu-server:~# chmod g+w /home/dev_dir
root@ubuntu-server:~#
```

Создаем каталог для совместной работы «/home/dev_dir/not_del_dir» (mkdir /home/dev_dir/not_del_dir). Меняем владельца папки (chown :developer /home/dev_dir/not_del_dir). Выдадим группе право на запись (chmod g+w /home/dev_dir/not_del_dir) и установим защиту от стирания всеми, кроме владельца (chmod +t /home/dev_dir/not_del_dir). В описании папки (ls -lha) для группы установилось «rws», а для удаления - в конце «t».

Под пользователем «dev_user1» доступ к файлу полный – редактируем (echo 'Теперь, dev_user1 успешно редактирует файл' > file_user1.txt) и удаляем (rm file_user1.txt).

```
root@ubuntu-server:/home/dev_dir - ssh - x
total 16K
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 .
drwxr-xr-x 3 root      root    4.0K Dec 31 13:25 ..
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 file_usr.txt
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 file_usr.txt
root@ubuntu-server:/home/dev_dir chmod g+rw /home/dev_dir
root@ubuntu-server:/home/dev_dir ls -la
total 16K
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 .
drwxr-xr-x 3 root      root    4.0K Dec 31 13:25 ..
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 file_usr.txt
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 file_usr.txt
root@ubuntu-server:/home/dev_dir
```

```
dev_user@ubuntu-server:/home/dev_dir/not_del_dir - ssh - x
try copyov & gsha mozhna dev_user2
try copyov & gsha mozhna dev_user2
dev_user@ubuntu-server:/home/dev_dir cd not_del_dir/
dev_user@ubuntu-server:/home/dev_dir/not_del_dir cat file_usr.txt
root gsha cozan dev_user!
dev_user@ubuntu-server:/home/dev_dir/not_del_dir echo "try copyov & gsha mozhn
nan dev_user? >> file_usr.txt
cat file_usr.txt: Permission denied
dev_user@ubuntu-server:/home/dev_dir/di not_del_dir ls -la
total 12K
drwxr-xr-x 2 root developer 4.0K Dec 31 06:53 .
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 ..
-rw-r--r- 1 dev_user! developer 41 Dec 31 06:53 file_usr.txt
dev_user@ubuntu-server:/home/dev_dir/not_del_dir nano file_usr.txt
dev_user@ubuntu-server:/home/dev_dir/not_del_dir ls -la
total 12K
drwxr-xr-x 2 root developer 4.0K Dec 31 07:02 .
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 ..
-rw-r--r- 1 dev_user! developer 41 Dec 31 06:53 file_usr.txt
dev_user@ubuntu-server:/home/dev_dir cd not_del_dir cat file_usr.txt
root gsha cozan dev_user!
dev_user@ubuntu-server:/home/dev_dir/not_del_dir cp file_usr.txt
cp: cannot remove '/file_usr.txt': Operation not permitted
dev_user@ubuntu-server:/home/dev_dir/not_del_dir
```

```
total 16K
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 .
drwxr-xr-x 3 root      root    4.0K Dec 31 13:25 ..
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 file_usr.txt
-rw-r--r- 1 dev_user! developer 100 Dec 31 13:50 not_del_dir
cd not_del_dir
ls -la
total 8,0K
drwxr-xr-x 2 root developer 4.0K Dec 31 07:11 .
drwxr-xr-x 3 root developer 4.0K Dec 31 06:42 ..
```

6. * 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.

Создаем каталог для совместной работы «/home/dev_dir/not_ls_dir» (mkdir /home/dev_dir/not_ls_dir). Выдадим всем право на запись и выполнение для всех (chmod 333 /home/dev_dir/not_ls_dir) и создадим несколько файлов.

Проверим достижение цели. Заходим под «dev_user2». При выполнении команды (ls -lha) – результат отрицательный - система выдает «Permission denied».

Но, мы знаем названия файлов и при выполнении (cat file1.txt) или (nano file2.txt), получаем доступ к их содержимому.

The image consists of three terminal window screenshots. The first window shows a root user on an Ubuntu server creating a directory /home/dev_dir/not_ls_dir and setting permissions to 333. The second window shows a dev_user2 user logging in and running 'ls -lha', which results in a 'Permission denied' error. The third window shows the same user running 'cat file1.txt' and 'cat file2.txt', which successfully displays the contents of the files.

```
root@ubuntu-server: /home/dev_dir/not_ls_dir
root@ubuntu-server: /home/dev_dir# mkdir /home/dev_dir/not_ls_dir
root@ubuntu-server: /home/dev_dir# chmod 333 /home/dev_dir/not_ls_dir
root@ubuntu-server: /home/dev_dir# cd not_ls_dir/
root@ubuntu-server: /home/dev_dir/not_ls_dir# echo 'Этот файл создан root' > file1.txt
root@ubuntu-server: /home/dev_dir/not_ls_dir# echo 'Этот файл создан root' > file2.txt
root@ubuntu-server: /home/dev_dir/not_ls_dir#
```

```
dev_user2@ubuntu-server: /home/dev_dir/not_ls_dir
System load: 0.0 Processes: 117
Usage of /: 48.5% of 8.79GB Users logged in: 1
Memory usage: 22% IPv4 address for enp0s3: 192.168.1.71
Swap usage: 0%

* Introducing self-healing high availability clusters in MicroK8s.
Simple, hardened, Kubernetes for production, from RaspberryPi to DC.
https://microk8s.io/high-availability

0 updates can be installed immediately.
0 of these updates are security updates.

Last login: Thu Dec 31 06:51:34 2020 from 192.168.1.127
dev_user2@ubuntu-server:~$ cd /home/dev_dir/not_ls_dir/
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$ ls -lha
ls: cannot open directory '.': Permission denied
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$
```

```
dev_user2@ubuntu-server: /home/dev_dir/not_ls_dir
System information as of Thu Dec 31 18:55:14 UTC 2020

System load: 0.0 Processes: 117
Usage of /: 48.5% of 8.79GB Users logged in: 1
Memory usage: 22% IPv4 address for enp0s3: 192.168.1.71
Swap usage: 0%

* Introducing self-healing high availability clusters in MicroK8s.
Simple, hardened, Kubernetes for production, from RaspberryPi to DC.
https://microk8s.io/high-availability

0 updates can be installed immediately.
0 of these updates are security updates.

Last login: Thu Dec 31 06:51:34 2020 from 192.168.1.127
dev_user2@ubuntu-server:~$ cd /home/dev_dir/not_ls_dir/
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$ ls -lha
ls: cannot open directory '.': Permission denied
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$ cat file1.txt
Этот файл создан root
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$ cat file2.txt
Этот файл создан root
dev_user2@ubuntu-server:/home/dev_dir/not_ls_dir$
```