

# Отчёт по лабораторной работе 3

## Настройка прав доступа

Цвелев С.А. НПИбд-02-22

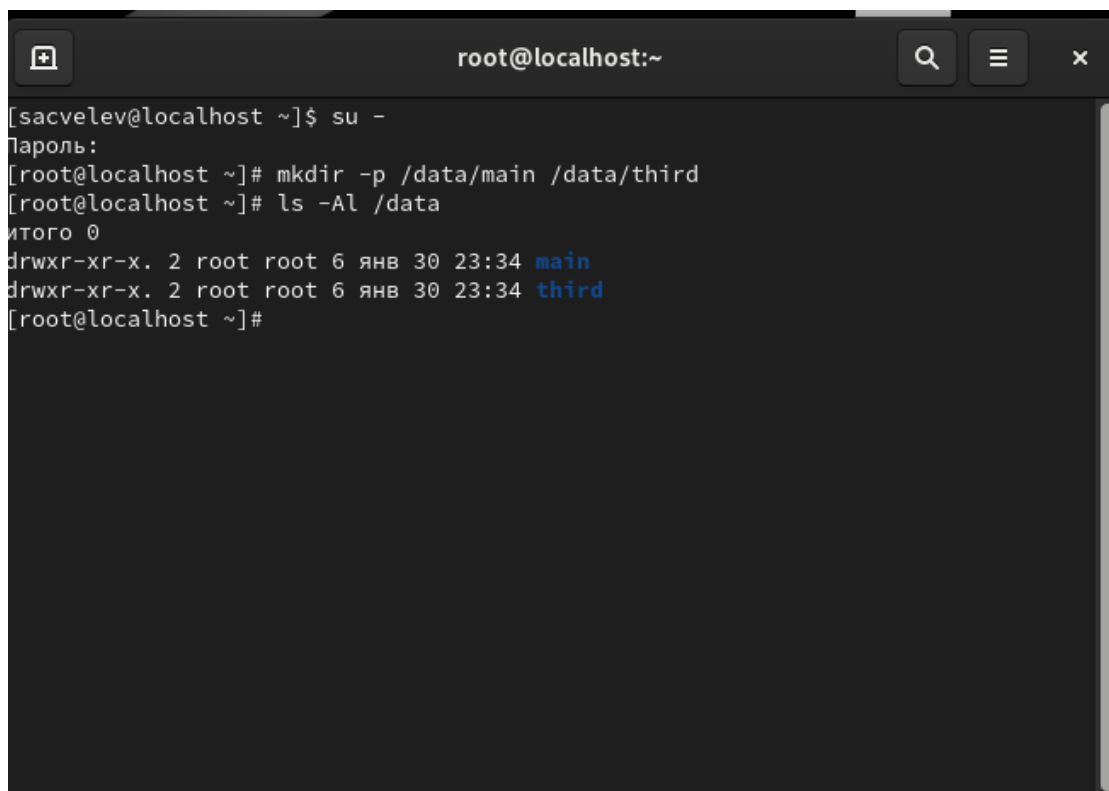
### Содержание

#### 1 Цель работы

Получение навыков настройки базовых и специальных прав доступа для групп пользователей в операционной системе типа Linux.

#### 2 Выполнение лабораторной работы

Открываем терминал с учётной записью root, создаём каталоги /data/main и /data/third. Проверяем, кто является их владельцами.



```
root@localhost:~  
[sacvelev@localhost ~]$ su -  
Пароль:  
[root@localhost ~]# mkdir -p /data/main /data/third  
[root@localhost ~]# ls -Al /data  
итого 0  
drwxr-xr-x. 2 root root 6 янв 30 23:34 main  
drwxr-xr-x. 2 root root 6 янв 30 23:34 third  
[root@localhost ~]#
```

Затем ставим владельцев на main и third, затем устанавливаем им разрешения, позволяющие записывать владельцам файлы в эти каталоги и запрещающие доступ другим пользователям и группам.

```
root@localhost:~  
[sacvelev@localhost ~]$ su -  
Пароль:  
[root@localhost ~]# chgrp main /data/main  
[root@localhost ~]# chgrp third /data/third  
[root@localhost ~]# chmod 770 /data/main  
[root@localhost ~]# chmod 770 /data/third  
[root@localhost ~]# ls -Al data  
ls: невозможно получить доступ к 'data': Нет такого файла или каталога  
[root@localhost ~]# ls -Al /data  
итого 0  
drwxrwx---. 2 root main 6 янв 30 23:34 main  
drwxrwx---. 2 root third 6 янв 30 23:34 third  
[root@localhost ~]#
```

В другом терминале заходим под пользователем bob и, перейдя в каталог /data/main, создаем emptyfile. Получилось. Повторяем то же самое с каталогом /data/third и видим ошибку доступа. Вспоминаем, что в прошлой работе мы назначили bob в группу доступа main, в то время как third находится в отдельной. Вот поэтому и не получилось добавить файл в каталоге third.

```
bob@localhost:~  
root@localhost:~ x bob@localhost:~ x alice@localhost:/d... x  
[sacvelev@localhost ~]$ su - bob  
Пароль:  
[bob@localhost ~]$ cd /data/main  
[bob@localhost main]$ touch emptyfile  
[bob@localhost main]$ ls -Al  
итого 0  
-rw-r--r--. 1 bob bob 0 янв 30 23:46 emptyfile  
[bob@localhost main]$ cd  
[bob@localhost ~]$ cd /data/third  
-bash: cd: /data/third: Отказано в доступе  
[bob@localhost ~]$
```

В другом терминале заходим под пользователем alice и создаем в /data/main файлы alice1 и alice2

```
alice@localhost:/data/main

[sacvelev@localhost ~]$ su - alice
Пароль:
[alice@localhost ~]$ cd /data/main
[alice@localhost main]$ touch alice1
[alice@localhost main]$ touch alice2
[alice@localhost main]$
```

Переходим на пользователя bob и, перейдя в каталог /data/main, удаляем все файлы, принадлежащие пользователю alice (само собой, проверяем). После, создаём файлы bob1 и bob2

```
bob@localhost:/data/main

[sacvelev@localhost ~]$ su - bob
Пароль:
[bob@localhost ~]$ cd /data/main
[bob@localhost main]$ ls -l
итого 0
-rw-r--r--. 1 alice alice 0 янв 30 23:56 alice1
-rw-r--r--. 1 alice alice 0 янв 30 23:56 alice2
-rw-r--r--. 1 bob bob 0 янв 30 23:46 emptyfile
[bob@localhost main]$ rm -f alice*
[bob@localhost main]$ ls -l
итого 0
-rw-r--r--. 1 bob bob 0 янв 30 23:46 emptyfile
[bob@localhost main]$ touch bob1
[bob@localhost main]$ touch bob2
[bob@localhost main]$
```

Под пользователем root устанавливаю для /data/main бит идентификатора группы и sticky-бит для общего каталога группы. Теперь, под пользователем alice создаю там файлы alice3 и alice4, что принадлежат группе main. Пробую удалить этим пользователем файлы пользователя bob и убеждаюсь, что это не удаётся.

```
alice@localhost:/data/main

[sacvelev@localhost ~]$ su -
Пароль:
[root@localhost ~]# chmod g+s,o+t /data/main
[root@localhost ~]# su alice
[alice@localhost root]$ cd /data/main
[alice@localhost main]$ touch alice3
[alice@localhost main]$ touch alice4
[alice@localhost main]$ ls -l
итого 0
-rw-r--r--. 1 alice main 0 янв 31 00:28 alice3
-rw-r--r--. 1 alice main 0 янв 31 00:28 alice4
-rw-r--r--. 1 bob   bob   0 янв 31 00:01 bob1
-rw-r--r--. 1 bob   bob   0 янв 31 00:01 bob2
-rw-r--r--. 1 bob   bob   0 янв 30 23:46 emptyfile
[alice@localhost main]$ su bob
Пароль:
[bob@localhost main]$ su alice
Пароль:
[alice@localhost main]$ rm -rf bob*
rm: невозможно удалить 'bob1': Операция не позволена
rm: невозможно удалить 'bob2': Операция не позволена
[alice@localhost main]$
```

Устанавливаем права на чтение и выполнение в каталогах /data/main и /data/third для групп third и main. Используем getfacl, чтобы убедиться.

```
root@localhost:~

[root@localhost ~]# setfacl -m g:third:rx /data/main
[root@localhost ~]# setfacl -m g:main:rx /data/third
[root@localhost ~]# getfacl /data/main
getfacl: Removing leading '/' from absolute path names
# file: data/main
# owner: root
# group: main
# flags: -st
user::rwx
group::rwx
group:third:r-x
mask::rwx
other::---

[root@localhost ~]# getfacl /data/third
getfacl: Removing leading '/' from absolute path names
# file: data/third
# owner: root
# group: third
user::rwx
group::rwx
group:main:r-x
mask::rwx
other::---
```

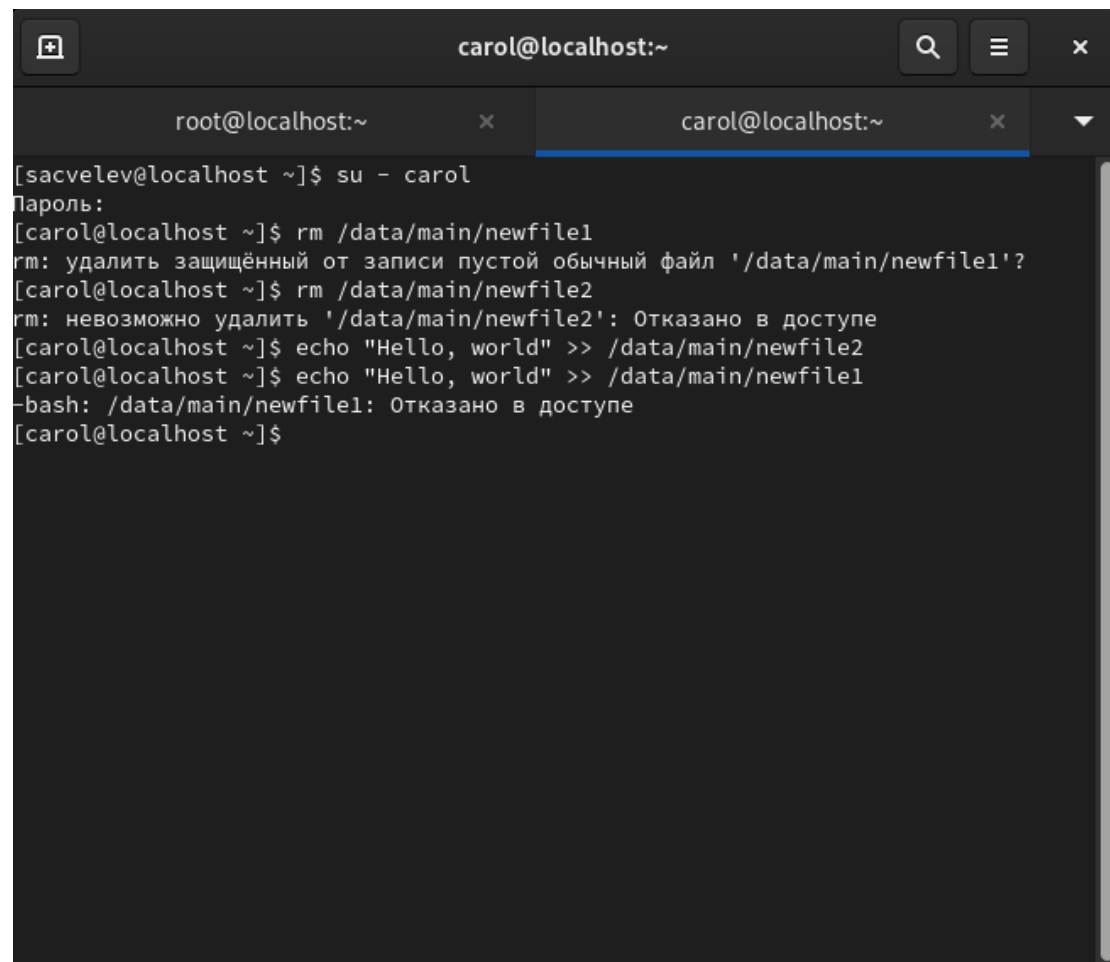
В каталогах /data/main и /data/third создаем файл newfile1. Затем проверяю полномочия с помощью getfacl. Как можно заметить, владельцем /data/third/newfile1 является группа root.

```
root@localhost:~  
[sacvelev@localhost ~]$ su -  
Пароль:  
[root@localhost ~]# touch /data/main/newfile1  
[root@localhost ~]# getfacl /data/main/newfile1  
getfacl: Removing leading '/' from absolute path names  
# file: data/main/newfile1  
# owner: root  
# group: main  
user::rw-  
group::r--  
other::r--  
  
[root@localhost ~]# touch /data/third/newfile1  
[root@localhost ~]# getfacl /data/third/newfile1  
getfacl: Removing leading '/' from absolute path names  
# file: data/third/newfile1  
# owner: root  
# group: root  
user::rw-  
group::r--  
other::r--  
  
[root@localhost ~]#
```

Устанавливаю ACL по умолчанию для каталогов /data/main и /data/third. Убеждаюсь, что настройки работают, добавив в /data/main файл newfile, и проверяю текущие полномочия.

```
root@localhost:~  
[sacvelev@localhost ~]$ su -  
Пароль:  
[root@localhost ~]# setfacl -m d:g:third:rw- /data/main  
[root@localhost ~]# setfacl -m d:g:main:rw- /data/third  
[root@localhost ~]# touch /data/main/newfile2  
[root@localhost ~]# getfacl /data/main/newfile2  
getfacl: Removing leading '/' from absolute path names  
# file: data/main/newfile2  
# owner: root  
# group: main  
user::rw-  
group::rw-  
group:third:rw-  
mask::rw-  
other::---  
#effective:rw-  
#effective:rw-  
  
[root@localhost ~]#
```

Вхожу под пользователем carol и проверяю операции с файлом. Пытаюсь удалить newfile1 в папке /data/main, а затем newfile2. С newfile1 ничего не получается, ибо он защищён от записи.



```
carol@localhost:~  
root@localhost:~ x carol@localhost:~ x  
[sacvelev@localhost ~]$ su - carol  
Пароль:  
[carol@localhost ~]$ rm /data/main/newfile1  
rm: удалить защищённый от записи пустой обычный файл '/data/main/newfile1'?  
[carol@localhost ~]$ rm /data/main/newfile2  
rm: невозможно удалить '/data/main/newfile2': Отказано в доступе  
[carol@localhost ~]$ echo "Hello, world" >> /data/main/newfile2  
[carol@localhost ~]$ echo "Hello, world" >> /data/main/newfile1  
-bash: /data/main/newfile1: Отказано в доступе  
[carol@localhost ~]$
```

### 3 Контрольные вопросы

1. Для установки владельца группы с использованием команды chown нужно указать опцию -R для рекурсивного применения изменений ко всему содержимому каталога. chown bob:main /data/third/newfile
2. Для поиска всех файлов, принадлежащих конкретному пользователю, можно использовать команду find. find ~ -user bob -print
3. Для установки разрешений на чтение, запись и выполнение для всех файлов в каталоге /data для пользователей и владельцев групп используется chmod. Пример: chmod 770 /data
4. Для добавления разрешения на выполнение используйте команду chmod с опцией +x. Пример: chmod +x file
5. Чтобы убедиться, что групповые разрешения для новых файлов сохраняют владельцу группы каталога, используется getfacl. Пример: getfacl "имя каталога"

6. Для ограничения удаления файлов только владельцами используется команда `chmod`. Пример: `chmod g+s,o+t /data/main`
7. Для добавления ACL и предоставления членам группы прав на чтение используется `setfacl`. Пример: `setfacl -m g:group:r <file/dir>`
8. Чтобы гарантировать разрешения для всех файлов и подкаталогов, используется опция `-dm` с `setfacl`. Пример: `setfacl -dm g:group:r /dir`
9. Чтобы установить `umask`, чтобы другие пользователи не получали разрешений, используется `007`.
10. Для предотвращения случайного удаления файла используется команда `chattr` с опцией `+i`. Пример: `chattr +i myfile`.