

Лабораторная работа №3

Настройка прав доступа

Комягин Андрей Николаевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Управление базовыми разрешениями	6
2.2	Управление специальными разрешениями	8
2.3	Управление расширенными разрешениями с использованием списков ACL	9
3	Контрольные вопросы	13
4	Вывод	15
	Список литературы	16

Список иллюстраций

2.1	владелец каталогов	6
2.2	изменение владельцев	6
2.3	установка разрешений	7
2.4	попытка редактирования main бобом	7
2.5	попытка редактирования third бобом	7
2.6	Создание файлов элис	8
2.7	удаление файлов элис с учетки боба	8
2.8	создание файлов боба	8
2.9	установка бит для каталога	9
2.10	работа с файлами и доступом	9
2.11	работа с разрешениями	10
2.12	проверка полномочий	10
2.13	аналогичные действия	11
2.14	проверка настроек ACL	11
2.15	настройки ACL	11
2.16	проверка полномочий пользователя группы third	12

Список таблиц

1 Цель работы

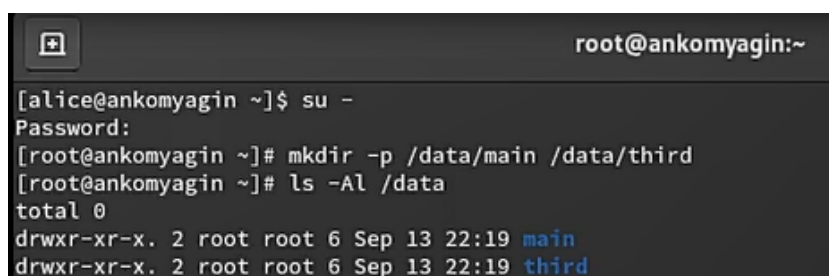
Получение навыков настройки базовых и специальных прав доступа для групп пользователей в операционной системе типа Linux.

2 Выполнение лабораторной работы

2.1 Управление базовыми разрешениями

Создим структуру каталогов с разными разрешениями доступа для разных групп пользователей.

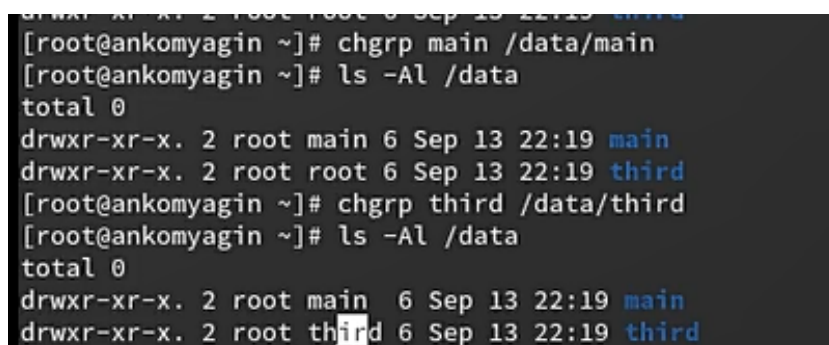
Откроем терминал с учётной записью root, создадим каталоги и посмотрим, кто является их владельцем (рис. 2.1).



```
root@ankomyagin:~  
[alice@ankomyagin ~]$ su -  
Password:  
[root@ankomyagin ~]# mkdir -p /data/main /data/third  
[root@ankomyagin ~]# ls -Al /data  
total 0  
drwxr-xr-x. 2 root root 6 Sep 13 22:19 main  
drwxr-xr-x. 2 root root 6 Sep 13 22:19 third
```

Рис. 2.1: владелец каталогов

Изменим владельцев этих каталогов с **root** на **main** и **third**(рис. 2.2).



```
[root@ankomyagin ~]# chgrp main /data/main  
[root@ankomyagin ~]# ls -Al /data  
total 0  
drwxr-xr-x. 2 root main 6 Sep 13 22:19 main  
drwxr-xr-x. 2 root root 6 Sep 13 22:19 third  
[root@ankomyagin ~]# chgrp third /data/third  
[root@ankomyagin ~]# ls -Al /data  
total 0  
drwxr-xr-x. 2 root main 6 Sep 13 22:19 main  
drwxr-xr-x. 2 root third 6 Sep 13 22:19 third
```

Рис. 2.2: изменение владельцев

Установим разрешения, позволяющие владельцам каталогов записывать файлы в эти каталоги и запрещающие доступ к содержимому каталогов всем другим пользователям и группам (рис. 2.3).

```
[root@ankomyagin ~]# chmod 770 /data/main
[root@ankomyagin ~]# chmod 770 /data/third
[root@ankomyagin ~]# ls -l /data/
total 0
drwxrwx---. 2 root main 6 Sep 13 22:19 main
drwxrwx---. 2 root third 6 Sep 13 22:19 third
```

Рис. 2.3: установка разрешений

В другом терминале перейдём под учётную запись пользователя bob и попробуем перейти в каталог **/data/main** и создать файл **emptyfile**. У нас получится это сделать, так как боб состоит в группе main, а права доступа == 770 (рис. 2.4).

```
[ankomyagin@ankomyagin ~]$ su - bob
Password:
su: Authentication failure
[ankomyagin@ankomyagin ~]$ su - bob
Password:
[bob@ankomyagin ~]$ cd /data/main
[bob@ankomyagin main]$ touch emptyfile
[bob@ankomyagin main]$ ls -Al
total 0
-rw-r--r--. 1 bob bob 0 Sep 13 22:23 emptyfile
```

Рис. 2.4: попытка редактирования main бобом

Под пользователем bob попробуем перейти в каталог **/data/third** и создать файл **emptyfile** в этом каталоге. У нас не получится даже перейти в каталог, так как боб не состоит в группе **third**

Создадим пользователя bob и посмотрим, в каких группах он состоит(рис. 2.5).

```
[bob@ankomyagin data]$ cd
main/ third/
[bob@ankomyagin data]$ cd third/
-bash: cd: third/: Permission denied
```

Рис. 2.5: попытка редактирования third бобом

2.2 Управление специальными разрешениями

Разберем работу со **sticky bit**

Откроем терминал под пользователем **alice** и создадим в каталоге **/data/main** два файла(рис. 2.6)

```
[ankomyagin@ankomyagin ~]$ su alice
Password:
[alice@ankomyagin ankomyagin]$ cd /data/main/
[alice@ankomyagin main]$ touch alice1
[alice@ankomyagin main]$ touch alice2
[alice@ankomyagin main]$
```

Рис. 2.6: Создание файлов элис

Перейдём под учётную запись пользователя **bob** и попытаемся удалить файлы пользователя **alice**. Убедимся, что они удалены (рис. 2.7)

```
[bob@ankomyagin data]$ ls
main third
[bob@ankomyagin data]$ cd main/
[bob@ankomyagin main]$ ls -l
total 0
-rw-r--r--. 1 alice alice 0 Sep 13 22:25 alice1
-rw-r--r--. 1 alice alice 0 Sep 13 22:25 alice2
-rw-r--r--. 1 bob bob 0 Sep 13 22:23 emptyfile
[bob@ankomyagin main]$ rm -f alice*
[bob@ankomyagin main]$ ls -l
total 0
-rw-r--r--. 1 bob bob 0 Sep 13 22:23 emptyfile
[bob@ankomyagin main]$
```

Рис. 2.7: удаление файлов элис с учетки боба

Создадим два файла с учетной записи пользователя **bob** (рис. 2.8).

```
[bob@ankomyagin main]$ touch bob1
[bob@ankomyagin main]$ touch bob2
[bob@ankomyagin main]$
```

Рис. 2.8: создание файлов боба

В терминале под пользователем root установим для каталога /data/main бит идентификатора группы, а также sticky-бит для разделяемого (общего) каталога группы (рис. 2.9)

```
[root@ankomyagin ~]# chmod g+s,o+t /data/main
[root@ankomyagin ~]#
```

Рис. 2.9: установка бит для каталога

Переключимся на учётную запись пользователя alice, создадим два файла. Видим, что они принадлежат группе main. Также попробуем удалить файлы, принадлежащие бобу. У нас это не выйдет, так как sticky-bit предотвратит удаление этих файлов пользователем alice, поскольку этот пользователь не является владельцем этих файлов (рис. 2.10)

```
[alice@ankomyagin main]$ touch alice3
[alice@ankomyagin main]$ touch alice4
[alice@ankomyagin main]$ ls -l
total 0
-rw-r--r--. 1 alice main 0 Sep 13 22:27 alice3
-rw-r--r--. 1 alice main 0 Sep 13 22:27 alice4
-rw-r--r--. 1 bob   bob   0 Sep 13 22:26 bob1
-rw-r--r--. 1 bob   bob   0 Sep 13 22:26 bob2
-rw-r--r--. 1 bob   bob   0 Sep 13 22:23 emptyfile
[alice@ankomyagin main]$ rm -rf bob*
rm: cannot remove 'bob1': Operation not permitted
rm: cannot remove 'bob2': Operation not permitted
[alice@ankomyagin main]$
```

Рис. 2.10: работа с файлами и доступом

2.3 Управление расширенными разрешениями с использованием списков ACL

Откроем терминал с учётной записью **root**, установим права на чтение и выполнение в каталогах и используем команду **getfacl**, чтобы убедиться в правильности установки разрешений (рис. 2.11)

```

[root@ankomyagin ~]# su -
[root@ankomyagin ~]# setfacl -m g:third:rx /data/main
[root@ankomyagin ~]# setfacl -m g:main:rx /data/third
[root@ankomyagin ~]# 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@ankomyagin ~]# 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:---
[root@ankomyagin ~]#

```

Рис. 2.11: работа с разрешениями

Создадим новый файл с именем **newfile1** в каталоге **/data/main**, проверим текущие назначенные ему полномочия. Люди из группы и остальные имеют только право на чтение (мы это назначили) (файл неисполняем) (рис. 2.12)

```

[root@ankomyagin ~]# touch /data/main/newfile1
[root@ankomyagin ~]# 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@ankomyagin ~]#

```

Рис. 2.12: проверка полномочий

Выполним аналогичные действия для каталога **/data/third**. Полномочия такие же, отличается только группа. (рис. 2.13)

```
[root@ankomyagin ~]# touch /data/third/newfile1
[root@ankomyagin ~]# 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@ankomyagin ~]#
```

Рис. 2.13: аналогичные действия

Установим ACL по умолчанию для каталогов и убедимся, что настройки работают, добавив новый файл в каталог. (рис. 2.14)

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

Рис. 2.14: проверка настроек ACL

Проведем аналогичные действия для каталога /data/third (рис. 2.15).

```
[root@ankomyagin ~]# touch /data/third/newfile2
[root@ankomyagin ~]# getfacl /data/third/newfile2
getfacl: Removing leading '/' from absolute path names
# file: data/third/newfile2
# owner: root
# group: root
user::rw-
group::rwx
group:main:rwx
mask::rw-
other::---
#effective:rwx
#effective:rwx
```

Рис. 2.15: настройки ACL

Для проверки полномочий группы **third** в каталоге **/data/third** войдём в другом

терминале под учётной записью члена группы **third** и проверим операции с файлами

Удалить файлы не получится, так как он не является владельцем, однако сможет внести изменения во второй файл, потому что мы создали его после изменения настроек полномочий(рис. 2.16)

```
[bob@ankomyagin main]$ su carol
Password:
[carol@ankomyagin main]$ rm /data/main/newfile1
rm: remove write-protected regular empty file '/data/main/newfile1'? yes
rm: cannot remove '/data/main/newfile1': Permission denied
[carol@ankomyagin main]$ rm /data/main/newfile2
rm: cannot remove '/data/main/newfile2': Permission denied
[carol@ankomyagin main]$ echo "Hello, world" >> /data/main/newfile1
bash: /data/main/newfile1: Permission denied
[carol@ankomyagin main]$ echo "Hello, world" >> /data/main/newfile2
[carol@ankomyagin main]$
```

Рис. 2.16: проверка полномочий пользователя группы third

3 Контрольные вопросы

1. Как следует использовать команду **chown**, чтобы установить владельца группы для файла?

Чтобы установить владельца и группу для файла, нужно использовать команду **chown** следующим образом:

chown :groupname filename

2. С помощью какой команды можно найти все файлы, принадлежащие конкретному пользователю?

Для этого можно использовать команду **find**:

find /path/to/search -user username

3. Как применить разрешения на чтение, запись и выполнение для всех файлов в каталоге **/data** для пользователей и владельцев групп, не устанавливая никаких прав для других?

Используем команду **chmod**:

chmod 770 /data/*

4. Какая команда позволяет добавить разрешение на выполнение для файла, который необходимо сделать исполняемым?

chmod +x filename

5. Какая команда позволяет убедиться, что групповые разрешения для всех новых файлов, создаваемых в каталоге, будут присвоены владельцу группы этого каталога?

`chmod g+s /path/to/directory`

6. Необходимо, чтобы пользователи могли удалять только те файлы, владельцами которых они являются, или которые находятся в каталоге, владельцами которого они являются. С помощью какой команды можно это сделать?

`chmod g+s,o+t /путь`

7. Какая команда добавляет ACL, который предоставляет членам группы права доступа на чтение для всех существующих файлов в текущем каталоге?

`setfacl -m g:groupname:r /path/to/directory/*`

8. Что нужно сделать для гарантии того, что члены группы получают разрешения на чтение для всех файлов в текущем каталоге и во всех его подкаталогах, а также для всех файлов, которые будут созданы в этом каталоге в будущем?

Использовать команду `setfacl` с флагом `-R` (рекурсивно)

`setfacl -R -m g:groupname:rX /path/to/directory`

9. Какое значение `umask` нужно установить, чтобы «другие» пользователи не получали какие-либо разрешения на новые файлы?

`umask 007`

10. Какая команда гарантирует, что никто не сможет удалить файл `myfile` случайно?

`chattr +i myfile`

4 Вывод

В ходе выполнения лабораторной работы я получил навыки настройки базовых и специальных прав доступа для групп пользователей в операционной системе типа Linux.

Список литературы

Туис, курс Администрирование операционных систем