Отчёт по лабораторной работе №3

Настройка прав доступа

Щемелев Илья Владимирович

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Ход выполнения

2.1 Управление базовыми разрешениями доступа

1. В терминале выполнен вход под учётной записью **root** с использованием команды su.
   После успешной аутентификации получены административные привилегии, необходимые для управления владельцами и правами доступа.

2. В корневом каталоге файловой системы созданы каталоги **/data/main** и **/data/third**.
   Проверка владельцев и групп показала, что по умолчанию оба каталога принадлежат пользователю и группе **root**.

3. Для разграничения прав доступа изменены группы-владельцы каталогов:
   каталог **/data/main** назначен группе **main**,
   каталог **/data/third** — группе **third**.
   Повторная проверка подтвердила корректное назначение групп.

4. Для обоих каталогов установлены права доступа **770**, что обеспечивает полный доступ владельцу и группе и запрещает доступ всем остальным пользователям.
   Проверка показала корректную установку разрешений.

```
ivschemelev@ivschemelev:~$ su
Password:
root@ivschemelev:/home/ivschemelev# mkdir -p /data/main /data/third
root@ivschemelev:/home/ivschemelev# ls -Al /data
total 0
drwxr-xr-x. 2 root root 6 Jan 15 12:23 main
drwxr-xr-x. 2 root root 6 Jan 15 12:23 third
root@ivschemelev:/home/ivschemelev# chgrp main /data/main/
root@ivschemelev:/home/ivschemelev# chgrp third /data/third/
root@ivschemelev:/home/ivschemelev# ls -Al /data
total 0
drwxr-xr-x. 2 root main  6 Jan 15 12:23 main
drwxr-xr-x. 2 root third 6 Jan 15 12:23 third
root@ivschemelev:/home/ivschemelev# chmod 770 /data/main
root@ivschemelev:/home/ivschemelev# chmod 770 /data/third/
root@ivschemelev:/home/ivschemelev# ls -Al /data
total 0
drwxrwx---. 2 root main  6 Jan 15 12:23 main
drwxrwx---. 2 root third 6 Jan 15 12:23 third
root@ivschemelev:/home/ivschemelev# su bob
bob@ivschemelev:/home/ivschemelev$ cd /data/main/
bob@ivschemelev:/data/main$ touch emptyfile
bob@ivschemelev:/data/main$ ls -Al
total 0
-rw-r--r--. 1 bob bob 0 Jan 15 12:25 emptyfile
bob@ivschemelev:/data/main$ cd /data/third/
bash: cd: /data/third/: Permission denied
bob@ivschemelev:/data/main$
```

Рис. 2.1: Установка базовых прав доступа

5. В другом терминале выполнен вход под пользователем **bob**, являющимся членом группы **main**.

   Пользователь успешно перешёл в каталог **/data/main** и создал файл **emptyfile**, так как группа **main** имеет право записи в данный каталог.

6. Попытка пользователя **bob** перейти в каталог **/data/third** завершилась ошибкой *Permission denied*.

   Это связано с тем, что пользователь не состоит в группе **third**, а доступ для остальных пользователей запрещён.

2.2 Управление специальными разрешениями (setgid и sticky bit)

7. Под пользователем **alice** в каталоге **/data/main** созданы файлы **alice1** и **alice2**.

   Созданные файлы принадлежат пользователю **alice** и его основной группе.

8. Под пользователем **bob** выполнена попытка удаления файлов **alice1** и **alice2**.

   Удаление прошло успешно, так как на данном этапе в каталоге отсутствовал sticky bit, а у группы **main** были права на запись.

9. Пользователь **bob** создал файлы **bob1** и **bob2**.

   После этого под пользователем **root** для каталога **/data/main** были установлены бит идентификатора группы (**setgid**) и **sticky bit**, что превратило каталог в разделяемый для группы.

```
alice@ivschemelev:/data/main$ cd /data/main
alice@ivschemelev:/data/main$ touch alice1
alice@ivschemelev:/data/main$ touch alice2
alice@ivschemelev:/data/main$
exit
bob@ivschemelev:/data/main$ ls -l
total 0
-rw-r--r--. 1 alice alice 0 Jan 15 12:28 alice1
-rw-r--r--. 1 alice alice 0 Jan 15 12:28 alice2
-rw-r--r--. 1 bob   bob   0 Jan 15 12:25 emptyfile
bob@ivschemelev:/data/main$ rm -f alice*
bob@ivschemelev:/data/main$ ls -l
total 0
-rw-r--r--. 1 bob bob 0 Jan 15 12:25 emptyfile
bob@ivschemelev:/data/main$ touch bob1
bob@ivschemelev:/data/main$ touch bob2
bob@ivschemelev:/data/main$
exit
root@ivschemelev:/home/ivschemelev# chmod g+s,o+t /data/main/
root@ivschemelev:/home/ivschemelev# su alice
alice@ivschemelev:/home/ivschemelev$ cd /data/main/
alice@ivschemelev:/data/main$ touch alice3
alice@ivschemelev:/data/main$ touch alice4
alice@ivschemelev:/data/main$ ls -l
total 0
-rw-r--r--. 1 alice main 0 Jan 15 12:29 alice3
-rw-r--r--. 1 alice main 0 Jan 15 12:29 alice4
-rw-r--r--. 1 bob   bob  0 Jan 15 12:28 bob1
-rw-r--r--. 1 bob   bob  0 Jan 15 12:28 bob2
-rw-r--r--. 1 bob   bob  0 Jan 15 12:25 emptyfile
alice@ivschemelev:/data/main$ rm -rf bob*
rm: cannot remove 'bob1': Operation not permitted
rm: cannot remove 'bob2': Operation not permitted
alice@ivschemelev:/data/main$
```

Рис. 2.2: Установка setgid и sticky bit

10. Пользователь **alice** создал файлы **alice3** и **alice4**.

    Проверка показала, что файлы автоматически унаследовали группу **main**, что является результатом действия бита **setgid**.

11. Попытка пользователя **alice** удалить файлы **bob1** и **bob2** завершилась неуда-

чей.

Sticky bit предотвратил удаление, так как пользователь не является владельцем этих файлов.

2.3 Управление расширенными разрешениями с использованием ACL

12. Под пользователем **root** установлены расширенные права доступа ACL:
    группе **third** предоставлены права чтения и выполнения в каталоге **/data/main**,
    группе **main** — права чтения и выполнения в каталоге **/data/third**.
    Проверка ACL подтвердила корректность настроек.

```
root@ivschemelev:/home/ivschemelev#
root@ivschemelev:/home/ivschemelev# setfacl -m g:third:rx /data/main
root@ivschemelev:/home/ivschemelev# setfacl -m g:main:rx /data/third
root@ivschemelev:/home/ivschemelev# 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@ivschemelev:/home/ivschemelev# 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@ivschemelev:/home/ivschemelev#
```

Рис. 2.3: Настройка ACL и проверка getfacl

13. В каталоге **/data/main** создан файл **newfile1**.

    Проверка показала, что файл получил только стандартные права доступа, так как ACL по умолчанию для каталога ещё не были заданы.

    Аналогичное поведение наблюдалось и в каталоге **/data/third**.

```
root@ivschemelev:/home/ivschemelev# touch /data/main/newfile1
root@ivschemelev:/home/ivschemelev# 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@ivschemelev:/home/ivschemelev# touch /data/third/newfile1
root@ivschemelev:/home/ivschemelev# 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@ivschemelev:/home/ivschemelev#
```

Рис. 2.4: Проверка ACL нового файла

14. Для каталогов заданы ACL по умолчанию, обеспечивающие наследование расширенных прав для новых файлов:
    для **/data/main** — для группы **third**,
    для **/data/third** — для группы **main**.

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

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

root@ivschemelev:/home/ivschemelev#
```

Рис. 2.5: Установка ACL по умолчанию

15. После создания файлов **newfile2** установлено, что они автоматически унаследовали заданные ACL по умолчанию.
    Проверка показала наличие соответствующих прав доступа у групп **main** и **third**.

16. Под пользователем **carol**, являющимся членом группы **third**, выполнена проверка доступа.
    Попытки удаления файлов завершились отказом в доступе, так как пользователь не является владельцем файлов и действует sticky bit.
    Запись в файлы также оказалась невозможной из-за отсутствия соответствующих прав в ACL.

```
root@ivschemelev:/home/ivschemelev#
root@ivschemelev:/home/ivschemelev# su carol
carol@ivschemelev:/home/ivschemelev$ rm /data/main/newfile1
rm: remove write-protected regular empty file '/data/main/newfile1'? y
rm: cannot remove '/data/main/newfile1': Permission denied
carol@ivschemelev:/home/ivschemelev$ rm /data/main/newfile2
rm: cannot remove '/data/main/newfile2': Permission denied
carol@ivschemelev:/home/ivschemelev$ echo "Hello world" >> /data/main/newfile1
bash: /data/main/newfile1: Permission denied
carol@ivschemelev:/home/ivschemelev$ echo "Hello world" >> /data/main/newfile2
carol@ivschemelev:/home/ivschemelev$
```

Рис. 2.6: Проверка прав пользователем carol

3 Контрольные вопросы

1. Для установки владельца группы для файла используется команда **chown** с указанием имени группы после двоеточия.
   Такой способ позволяет изменить только групповую принадлежность файла без изменения владельца-пользователя.
   Например, команда `chown :main file.txt` устанавливает группу **main** владельцем файла *file.txt*.

2. Для поиска всех файлов, принадлежащих конкретному пользователю, применяется команда **find** с параметром `-user`.
   Она позволяет рекурсивно просмотреть файловую систему или указанный каталог.
   Например, команда `find /home -user alice` найдёт все файлы пользователя **alice** в каталоге */home* и его подкаталогах.

3. Чтобы применить разрешения на чтение, запись и выполнение для владельца и группы для всех файлов в каталоге */data* и не устанавливать никаких прав для остальных пользователей, используются права **770**.
   Это обеспечивает полный доступ владельцу и группе и полностью запрещает доступ для категории «other».
   Например, `chmod -R 770 /data`.

4. Для добавления разрешения на выполнение файлу, который необходимо сделать исполняемым, используется команда **chmod** с символьным указанием права выполнения.

Например, `chmod +x script.sh` добавляет право выполнения для всех категорий пользователей.

5. Чтобы убедиться, что групповые разрешения для всех новых файлов, создаваемых в каталоге, будут присвоены владельцу группы этого каталога, используется установка бита идентификатора группы (**setgid**).
   Это достигается командой `chmod g+s каталог`.
   Например, `chmod g+s /data/main` гарантирует, что новые файлы унаследуют группу **main**.

6. Для обеспечения возможности удаления файлов только их владельцами или владельцем каталога используется **sticky bit**.
   Этот атрибут применяется к каталогу и предотвращает удаление файлов посторонними пользователями.
   Например, `chmod +t /data/main`.

7. Для добавления ACL, предоставляющего членам группы права чтения для всех существующих файлов в текущем каталоге, используется команда **setfacl**.
   Например, `setfacl -m g:third:r *` добавляет группе **third** право чтения для всех файлов в каталоге.

8. Чтобы гарантировать, что члены группы получат разрешения на чтение для всех файлов в текущем каталоге, во всех его подкаталогах, а также для файлов, которые будут созданы в будущем, необходимо:

   - рекурсивно задать ACL для существующих файлов и каталогов;
   - установить ACL по умолчанию для каталога.
     Например, `setfacl -R -m g:third:r .` и `setfacl -d -m g:third:r .`.

9. Чтобы пользователи из категории «other» не получали никаких разрешений на новые файлы, необходимо установить значение **umask 007**.
   В этом случае новые файлы будут создаваться без каких-либо прав для

остальных пользователей.

Например, `umask 007`.

10. Для гарантии того, что файл *myfile* не сможет быть удалён случайно, используется установка атрибута неизменяемости.

    Это выполняется командой **chattr** с параметром `+i`.

    Например, `chattr +i myfile`, после чего файл нельзя будет удалить или изменить до снятия атрибута.

4 Заключение

В ходе лабораторной работы были изучены и практически применены механизмы управления доступом в Linux: базовые права, специальные биты **setgid** и **sticky bit**, а также расширенные списки контроля доступа **ACL**.
Результаты показали, что комбинация данных механизмов позволяет гибко и безопасно организовывать совместный доступ пользователей к файловым ресурсам.