Лабораторная работа №3

Настройка прав доступа

Чекмарев Александр Дмитриевич | группа НПИбд 03-24

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 Выполнение лабораторной работы

2.1 Управление базовыми разрешениями

Откроем терминал с учётной записью root В корневом каталоге создадим каталоги /data/main и /data/third: **mkdir -p /data/main /data/third**

```
adchekmarev@adchekmarev:~$ su -
Пароль:
Последний вход в систему: Сб сен 13 15:53:30 MSK 2025 на pts/0
root@adchekmarev:~# mkdir -p /data/main /data/third
```

Рисунок 2.1: Создание каталогов с root

Посмотрим, кто является владельцем этих каталогов. Для этого используем: **ls -Al /data**

```
root@adchekmarev:~# ls -Al /data
итого 0
drwxr-xr-x. 2 root root 6 сен 19 22:20 main
drwxr-xr-x. 2 root root 6 сен 19 22:20 third
```

Рисунок 2.2: Просмотр информации о каталогах

Прежде чем устанавливать разрешения, изменим владельцев этих каталогов с root на main и third соответственно:

```
chgrp main /data/main
chgrp third /data/third
```

```
root@adchekmarev:~# chgrp main /data/main
root@adchekmarev:~# chgrp third /data/third
```

Рисунок 2.3: Изменение владельцев каталогов

Посмотрим, кто теперь является владельцем этих каталогов: **ls -Al /data**

```
root@adchekmarev:~# ls -Al /data
итого 0
drwxr-xr-x. 2 root main  6 сен 19 22:20 main
drwxr-xr-x. 2 root third 6 сен 19 22:20 third
```

Рисунок 2.4: Просмотр информации о каталогах

Установим разрешения, позволяющие владельцам каталогов записывать файлы в эти каталоги и запрещающие доступ к содержимому каталогов всем другим пользователям и группам:

```
chmod 770 /data/main
chmod 770 /data/third
```

Проверим установленные права доступа

```
root@adchekmarev:~# chmod 770 /data/main
root@adchekmarev:~# chmod 770 /data/third
root@adchekmarev:~# ls -Al /data
итого 0
drwxrwx---. 2 root main  6 сен 19 22:20 main
drwxrwx---. 2 root third 6 сен 19 22:20 third
```

Рисунок 2.5: Изменение разрешений и просмотр прав доступа

В другом терминале перейдем под учётную запись пользователя bob Попробуем перейти в каталог /data/main и создать файл emptyfile в этом каталоге:

```
cd /data/main
touch emptyfile
ls -Al
```

Создался файл с правами под учетную запись пользователя bob

```
adchekmarev@adchekmarev:~$ su - bob
Пароль:
```

Рисунок 2.6: Переход в учётную запись bob

```
bob@adchekmarev:~$ cd /data/main
bob@adchekmarev:/data/main$ touch emptyfile
bob@adchekmarev:/data/main$ ls -Al
итого 0
-rw-r--r--. 1 bob bob 0 сен 19 22:23 emptyfile
```

Рисунок 2.7: Создание файла под учётной записью bob

Под пользователем bob попробуем перейти в каталог /data/third и создать файл emptyfile в этом каталоге

```
bob@adchekmarev:/data/main$ cd /data/third
-bash: cd: /data/third: Отказано в доступе
```

Рисунок 2.8: Попытка перейти в каталог

Мы не можем перейти в данный каталог и создать файл, так как у bob'a нет прав к каталогу third. Он находится в группе main, а не third.

2.2 Управление специальными разрешениями

Откроем новый терминал под пользователем alice. Перейдем в каталог /data/main: **cd /data/main** Создадим два файла, владельцем которых является alice:

```
touch alice1
touch alice2
```

```
adchekmarev@adchekmarev:~$ su - alice
Пароль:
Последний вход в систему: Сб сен 13 15:47:56 MSK 2025 на pts/0
alice@adchekmarev:~$ cd /data/main
alice@adchekmarev:/data/main$ touch alice1
alice@adchekmarev:/data/main$ touch alice2
```

Рисунок 2.9: Создание файлов под учётной записью alice в каталоге /data/main

В другом терминале перейдем под учётную запись пользователя bob Перейдем в каталог /data/main и введем: **ls -l**

```
bob@adchekmarev:/data/main$ ls -l
итого 0
-rw-r--r--. 1 alice alice 0 сен 19 22:27 alice1
-rw-r--r--. 1 alice alice 0 сен 19 22:27 alice2
-rw-r--r--. 1 bob   bob   0 сен 19 22:23 emptyfile
```

Рисунок 2.10: Просмотр информации о каталоге

Мы увидим два файла, созданные пользователем alice. Попробуем удалить файлы, принадлежащие пользователю alice: **rm -f alice*** Убедимся, что файлы будут удалены пользователем bob.

```
bob@adchekmarev:/data/main$ rm -f alice*
bob@adchekmarev:/data/main$ ls -l
итого 0
-rw-r--r--. 1 bob bob 0 сен 19 22:23 emptyfile
```

Рисунок 2.11: Удаление файлов alice и просмотр информации каталога

Создадим два файла, которые принадлежат пользователю bob:

```
touch bob1
touch bob2
```

```
bob@adchekmarev:/data/main$ touch bob1
bob@adchekmarev:/data/main$ touch bob2
```

Рисунок 2.12: Создание файлов под учётной записью bob

В терминале под пользователем root установим для каталога /data/main бит идентификатора группы, а также stiky-бит для разделяемого (общего) каталога группы: **chmod g+s,o+t /data/main**

```
root@adchekmarev:~# chmod g+s,o+t /data/main
```

Рисунок 2.13: Изменение бит индентификатора группы и sticky-бит для общего каталога группы

В терминале под пользователем alice создадим в каталоге /data/main файлы alice3 и alice4:

```
touch alice3
touch alice4
ls -l
```

Теперь мы должны увидеть, что два созданных нами файла принадлежат группе main, которая является группой-владельцем каталога /data/main.

```
alice@adchekmarev:/data/main$ touch alice3
alice@adchekmarev:/data/main$ touch alice4
alice@adchekmarev:/data/main$ ls -l
итого 0
-rw-r--r--. 1 alice main 0 сен 19 22:28 alice3
-rw-r--r--. 1 alice main 0 сен 19 22:28 alice4
-rw-r--r--. 1 bob   bob  0 сен 19 22:28 bob1
-rw-r--r--. 1 bob   bob  0 сен 19 22:28 bob2
-rw-r--r--. 1 bob   bob  0 сен 19 22:23 emptyfile
```

Рисунок 2.14: Создание файлов и просмотр информации о них

В терминале под пользователем alice попробуем удалить файлы, принадлежащие пользователю bob: **rm -rf bob***

```
alice@adchekmarev:/data/main$ rm -rf bob*
rm: невозможно удалить 'bob1': Операция не позволена
rm: невозможно удалить 'bob2': Операция не позволена
```

Рисунок 2.15: Попытка удаление файлов

Как мы видим sticky-bit предотвратил удаление этих файлов пользователем alice, поскольку этот пользователь не является владельцем этих файлов. Обратим внимание: поскольку пользователь alice является владельцем каталога /data/main, то он может удалить все свои файлы в любом случае.

2.3 Управление расширенными разрешениями с использованием списков ACL

Продолжим работать в созданных ранее каталогах /data/main и /data/third. Ранее для группы main были установлены разрешения на каталог /data/main, а у группы third — на каталог /data/third. Требуется установить для группы third разрешения на чтение в каталоге /data/main, а для группы main — разрешения на чтение в каталоге /data/third. Затем требуется установить права доступа по умолчанию, чтобы убедиться в правильности установки разрешений для новых элементов этих каталогов. Для этого будет использоваться пакет acl и команды setfacl (для установки прав) и getfacl (для просмотра установленных прав). Кратко опишем синтаксис команды setfacl.

Установить разрешения для пользователя: **setfacl -m «u:user:permissions» <file/dir>**

Установить разрешения для группы: **setfacl -m «g:group:permissions» <file/dir>**

Наследование записи ACL родительского каталога: **setfacl -dm «entry» <dir>**

Удаление записи ACL: **setfacl -x «entry» <file/dir>**

Синтаксис команды getfacl: **getfacl <file/dir>**

Применим команды setfacl и getfacl для выполнения поставленной задачи. Откроем терминал с учётной записью root Установим права на чтение и выполнение в каталоге /data/main для группы third и права на чтение и выполнение для группы main в каталоге /data/third:

```
setfacl -m g:third:rx /data/main
setfacl -m g:main:rx /data/third
```

```
root@adchekmarev:~# setfacl -m g:third:rx /data/main
root@adchekmarev:~# setfacl -m g:main:rx /data/third
```

Рисунок 2.16: Изменение прав каталогов для групп

Используем команду getfacl, чтобы убедиться в правильности установки разрешений:

```
getfacl /data/main
getfacl /data/third
```

```
root@adchekmarev:~# 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@adchekmarev:~# 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::---
```

Рисунок 2.17: Просмотр информации о каталогах

Создадим новый файл с именем newfile1 в каталоге /data/main: **touch /data/main/newfile1** Используем **getfacl /data/main/newfile1** для проверки текущих назначений полномочий.

```
root@adchekmarev:~# touch /data/main/newfile1
root@adchekmarev:~# 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--
```

Рисунок 2.18: Создание файла и просмотр информации о нём

Выполним аналогичные действия для каталога /data/third Создадим новый файл с именем newfile1 в каталоге /data/third и просмотрим информацию

```
root@adchekmarev:~# touch /data/third/newfile1
root@adchekmarev:~# 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--
```

Рисунок 2.19: Создание файла и просмотр информации о нём

Установим ACL по умолчанию для каталога /data/main: **setfacl -m d:g:third:rwx /data/main**

Добавим ACL по умолчанию для каталога /data/third: **setfacl -m d:g:main:rwx /data/third**

```
root@adchekmarev:~# setfacl -m d:g:third:rwx /data/main
root@adchekmarev:~# setfacl -m d:g:main:rwx /data/third
```

Рисунок 2.20: Изменение ACL по умолчанию для каталогов

Убедимся, что настройки ACL работают, добавив новый файл в каталог /data/main: **touch /data/main/newfile2**

Используем **getfacl /data/main/newfile2** для проверки текущих назначений полномочий

```
root@adchekmarev:~# touch /data/main/newfile2
root@adchekmarev:~# 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::---
```

Рисунок 2.21: Создание файла и просмотр информации

Выполним аналогичные действия для каталога /data/third. Создадим новый файл newfile2 в каталог /data/third и проверим текущие назначения полномочий

```
root@adchekmarev:~# touch /data/third/newfile2
root@adchekmarev:~# 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::---
```

Рисунок 2.22: Создание файла и просмотр информации

Для проверки полномочий группы third в каталоге /data/third войдем в другом терминале под учётной записью члена группы third: **su - carol**

Проверим операции с файлами:

```
rm /data/main/newfile1
rm /data/main/newfile2
```

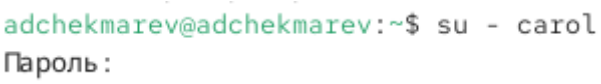

Рисунок 2.23: Вход в учётную запись carol

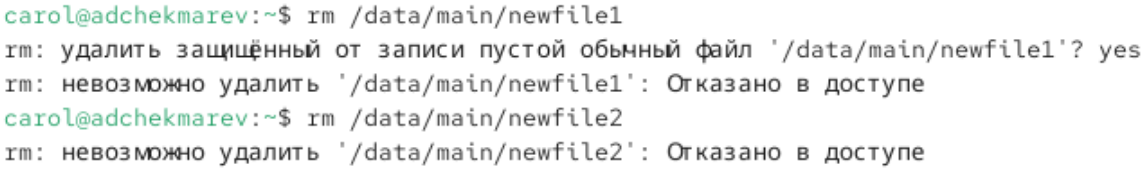

Рисунок 2.24: Попытка удаление файлов под учётной записью carol

Удалить файлы мы не смогли, так как newfile1 принадлежит пользователю root и группе main, newfile2 также принадлежит root, main и third, хоть и carol находится в последней группе, у группы недостаточно прав.

Проверим, возможно ли осуществить запись в файл:

```
echo ”Hello, world” >> /data/main/newfile1
echo ”Hello, world” >> /data/main/newfile2
```

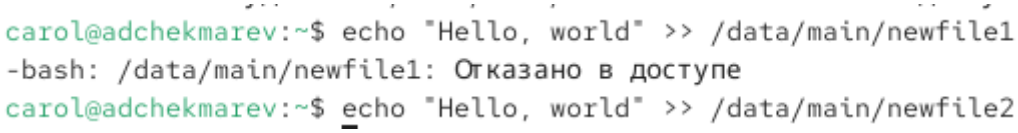

Рисунок 2.25: Попытка осуществить запись в файлы под учётной записью carol

Мы не смогли осуществить запись в newfile1, так как прав у группы third на нее нет. А вот уже в newfile2 все получилось, так как права на изменение файла у данной группы есть.

3 Контрольные вопросы

1. Как следует использовать команду chown, чтобы установить владельца группы для файла? Приведите пример. Создадим файл newfile.txt Узнаем информацию о нем

```
adchekmarev@adchekmarev:~/test$ ls -al
итого 12
drwxr-xr-x.  3 adchekmarev adchekmarev   88 сен 20 01:06 .
drwx------. 21 adchekmarev adchekmarev 4096 сен 19 21:57 ..
-rw-r--r--.  1 adchekmarev adchekmarev    0 сен 20 01:06 newfile.txt
```

Рисунок 3.1: Просмотр информации о файле

Изменим владельца и группу с правами root

```
root@adchekmarev:/home/adchekmarev/test# chown bob:bob newfile.txt
root@adchekmarev:/home/adchekmarev/test# ls -al
итого 12
drwxr-xr-x.  3 adchekmarev adchekmarev   88 сен 20 01:06 .
drwx------. 21 adchekmarev adchekmarev 4096 сен 19 21:57 ..
-rw-r--r--.  1 bob         bob            0 сен 20 01:06 newfile.txt
```

Рисунок 3.2: Изменение владельца и группы файла

2. С помощью какой команды можно найти все файлы, принадлежащие конкретному пользователю? Приведите пример. Воспользуемся командой **find / -user username** Просмотрим файлы carol в /home

```
root@adchekmarev:~# find /home -user carol
/home/carol
/home/carol/.mozilla
/home/carol/.mozilla/extensions
/home/carol/.mozilla/plugins
/home/carol/.bash_logout
/home/carol/.bash_profile
/home/carol/.bashrc
/home/carol/Pictures
/home/carol/Documents
/home/carol/.bash_history
```

Рисунок 3.3: Просмотр файлов принадлежащих carol

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? Приведите пример.

Можно воспользоваться этой командой **chmod 770 /data/***

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

Данная команда позволяет добавить разрешение **chmod +x script.sh**

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

Нужно воспользоваться командой **chmod g+s /название каталога** Конкретно за понимание изменения отвечает S, в rwx

```
root@adchekmarev:/home/adchekmarev/test# chmod g+s /test
root@adchekmarev:/home/adchekmarev/test# ls -ld /test
-rw-r-Sr--. 1 root root 0 сен 20 01:22 /test
```

Рисунок 3.4: Пример использования команды и проверка

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? Приведите пример.

Можно воспользоваться этой командой **chmod +t /название каталога** Где T - означает что изменения прошли

```
root@adchekmarev:/home/adchekmarev/test# chmod +t /test
root@adchekmarev:/home/adchekmarev/test# ls -ld /test
-rw-r-Sr-T. 1 root root 0 сен 20 01:22 /test
```

Рисунок 3.5: Пример использования команды и проверка

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

С помощью этой команды **setfacl -m g:groupname:r– ***

8. Что нужно сделать для гарантии того, что члены группы получат 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? Приведите пример.

Дать права группе на все существующие файлы и каталоги:
setfacl -R -m g:название группы:r– /название каталога
Настроить права по умолчанию для будущих файлов:
setfacl -R -d -m g:название группы:r– /название каталога

```
root@adchekmarev:/home/adchekmarev/test# setfacl -R -m g:main:r-- /test
root@adchekmarev:/home/adchekmarev/test# setfacl -R -d -m g:main:r-- /test
root@adchekmarev:/home/adchekmarev/test# getfacl /test
getfacl: Removing leading '/' from absolute path names
# file: test
# owner: root
# group: root
# flags: -st
user::rw-
group::r--
group:main:r--
mask::r--
other::r--
```

Рисунок 3.6: Пример использования команды и проверка

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

Нужно поставить значение 007

0 (для владельца) → не убираем ничего, оставляем все права

0 (для группы) → тоже оставляем все права

7 (для других) → убираем чтение 4 + запись 2 + выполнение 1 = 7

```
root@adchekmarev:/home/adchekmarev/test# umask 007
root@adchekmarev:/home/adchekmarev/test# touch newfile007
root@adchekmarev:/home/adchekmarev/test# mkdir test007
root@adchekmarev:/home/adchekmarev/test# ls -l
итого 8
-rw-rw----. 1 root        adchekmarev    0 сен 20 01:46 newfile007
-rw-r--r--. 1 root        adchekmarev    0 сен 20 01:23 newfiletest
-rw-r--r--. 1 bob         bob            0 сен 20 01:06 newfile.txt
-rw-r--r--. 1 adchekmarev adchekmarev 1360 сен  6 08:20 presentation.qmd
-rw-r--r--. 1 adchekmarev adchekmarev  641 сен  6 08:18 presentation.yml
drwxrws---. 2 root        adchekmarev    6 сен 20 01:46 test007
```

Рисунок 3.7: Пример использования команды и проверка

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

Если только запретить удаление, но оставить доступ:

chmod 555 myfile

Если полностью запретить доступ, то:

sudo chattr +i myfile

4 Выводы

Приобретены умения по управлению базовыми и специальными правами доступа для групп пользователей в ОС Linux.

Список литературы