

Постановка задачи

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

Выполнение работы

Управление базовыми разрешениями

1. Откройте терминал с учётной записью root:

```
su -
```

2. В корневом каталоге создайте каталоги /data/main и /data/third:

```
mkdir -p /data/main /data/third
```

Посмотрите, кто является владельцем этих каталогов. Для этого используйте:

```
ls -Al /data
```

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

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

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

Посмотрите, кто теперь является владельцем этих каталогов:

```
ls -Al /data
```

4. Установите 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
```

Проверьте установленные права доступа.

```
[zashikhalievaa@localhost ~]$ sudo i-
[sudo] пароль для zashikhalievaa:
sudo: i-: command not found
[zashikhalievaa@localhost ~]$ sudo -i
[root@localhost ~]# fdisk /dev/sdb -l
fdisk: невозможно открыть /dev/sdb: Нет такого файла или каталога
[root@localhost ~]# mkdir -p /data/main /data/third
[root@localhost ~]# ls -Al /data
итого 0
drwxr-xr-x. 2 root root 6 фев 16 12:46 main
drwxr-xr-x. 2 root root 6 фев 16 12:46 third
[root@localhost ~]# chgrp main /data/main
[root@localhost ~]# chgrp third /data/third/
[root@localhost ~]# ls -Al /data
итого 0
drwxr-xr-x. 2 root main 6 фев 16 12:46 main
drwxr-xr-x. 2 root third 6 фев 16 12:46 third
[root@localhost ~]# chmod 770 /data/main/
[root@localhost ~]# chmod 770 /data/third/
[root@localhost ~]# ls -Al /data
итого 0
drwxrwx---. 2 root main 6 фев 16 12:46 main
drwxrwx---. 2 root third 6 фев 16 12:46 third
[root@localhost ~]#
```

5. В другом терминале перейдите под учётную запись пользователя bob:

```
su – bob
```

6. Под пользователем bob попробуйте перейти в каталог /data/main и создать файл emptyfile в этом каталоге:

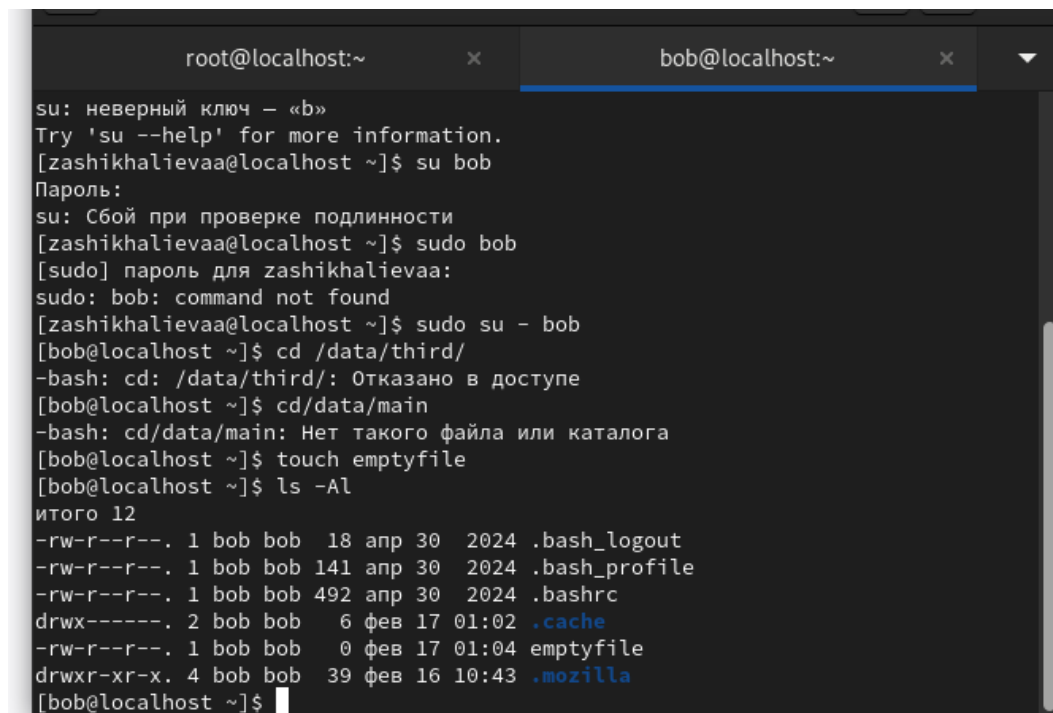
```
cd /data/main
```

```
touch emptyfile
```

```
ls -Al
```

Опишите и поясните результат этого действия.

7. Под пользователем bob попробуйте перейти в каталог /data/third и создать файл emptyfile в этом каталоге. Опишите и поясните результат этого действия.



```
root@localhost:~ x bob@localhost:~ x
su: неверный ключ - «b»
Try 'su --help' for more information.
[zashikhalieva@localhost ~]$ su bob
Пароль:
su: Сбой при проверке подлинности
[zashikhalieva@localhost ~]$ sudo bob
[sudo] пароль для zashikhalieva:
sudo: bob: command not found
[zashikhalieva@localhost ~]$ sudo su - bob
[bob@localhost ~]$ cd /data/third/
-bash: cd: /data/third/: Отказано в доступе
[bob@localhost ~]$ cd /data/main
-bash: cd /data/main: Нет такого файла или каталога
[bob@localhost ~]$ touch emptyfile
[bob@localhost ~]$ ls -Al
итого 12
-rw-r--r--. 1 bob bob 18 апр 30 2024 .bash_logout
-rw-r--r--. 1 bob bob 141 апр 30 2024 .bash_profile
-rw-r--r--. 1 bob bob 492 апр 30 2024 .bashrc
drwx-----. 2 bob bob 6 фев 17 01:02 .cache
-rw-r--r--. 1 bob bob 0 фев 17 01:04 emptyfile
drwxr-xr-x. 4 bob bob 39 фев 16 10:43 .mozilla
[bob@localhost ~]$
```

Bob является членом группы main, и у него есть доступ к каталогу /data/main, так как этот каталог также принадлежит группе main. Для данного каталога установлены права доступа 770, что означает, что владельцы файлов и члены группы могут читать, записывать и выполнять действия с файлами в этом каталоге (полные права — 7: rwx). Все остальные пользователи, не являющиеся владельцами или членами группы, не имеют никаких прав на доступ (0: ---).

В случае с каталогом /data/third, который принадлежит группе third, права доступа также установлены как 770. Это означает, что члены группы third и владелец каталога могут выполнять любые действия с файлами (7: rwx). Однако Bob не является членом группы third, поэтому для него действуют права "для остальных пользователей", что в данном случае равно 0. Это означает, что у него нет никаких прав на доступ к каталогу /data/third — он не может ни читать, ни изменять, ни выполнять файлы в этом каталоге.

Управление специальными разрешениями

1. Откройте новый терминал под пользователем alice.

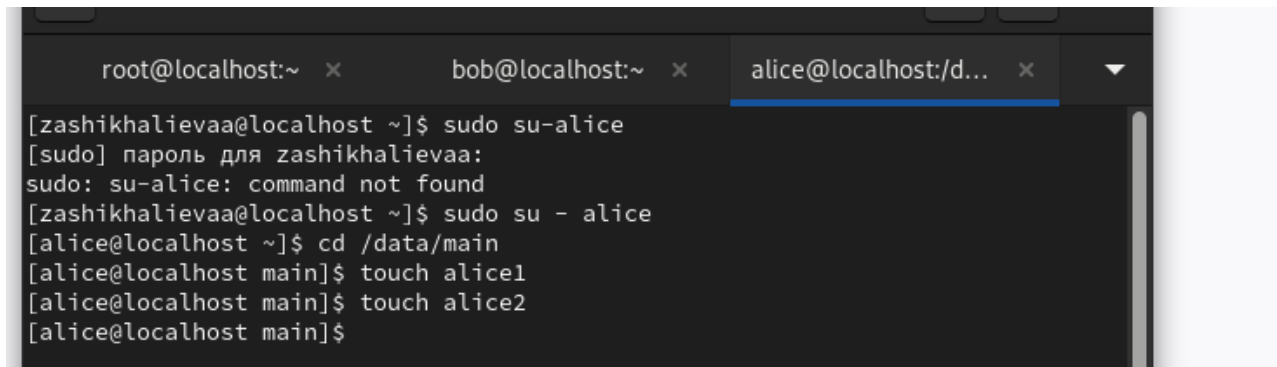
2. Перейдите в каталог /data/main:

```
cd /data/main
```

Создайте два файла, владельцем которых является alice:

```
touch alice1
```

```
touch alice2
```



The screenshot shows a terminal window with three tabs: 'root@localhost:~', 'bob@localhost:~', and 'alice@localhost:/d...'. The active tab is 'alice@localhost:/d...'. The terminal output shows the following commands and responses:

```
[zashikhalievaa@localhost ~]$ sudo su-alice
[sudo] пароль для zashikhalievaa:
sudo: su-alice: command not found
[zashikhalievaa@localhost ~]$ sudo su - alice
[alice@localhost ~]$ cd /data/main
[alice@localhost main]$ touch alice1
[alice@localhost main]$ touch alice2
[alice@localhost main]$
```

3. В другом терминале перейдите под учётную запись пользователя bob (пользователь bob является членом группы main, как и alice):

```
su - bob
```

4. Перейдите в каталог /data/main:

```
cd /data/main
```

и в этом каталоге введите:

```
ls -l
```

Вы увидите два файла, созданные пользователем alice. Попробуйте удалить файлы, принадлежащие пользователю alice:

```
rm -f alice*
```

Убедитесь, что файлы будут удалены пользователем bob.

5. Создайте два файла, которые принадлежат пользователю bob:

```
touch bob1
touch bob2
```

```
[bob@localhost ~]$ sudo su alice
[sudo] пароль для bob:
bob is not in the sudoers file. This incident will be reported.
[bob@localhost ~]$ cd /data/main
[bob@localhost main]$ ls -l
итого 0
-rw-r--r--. 1 alice alice 0 фев 17 01:08 alice1
-rw-r--r--. 1 alice alice 0 фев 17 01:08 alice2
[bob@localhost main]$ rm -f alice*
[bob@localhost main]$ ls -l
итого 0
[bob@localhost main]$ touch bob1
[bob@localhost main]$ touch bob2
[bob@localhost main]$
```

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

```
[-h host] [-p prompt] [-R directory] [-T timeout] [-u user] file
...
[root@localhost ~]# chmod g+s,o+t /data/main
[root@localhost ~]#
```

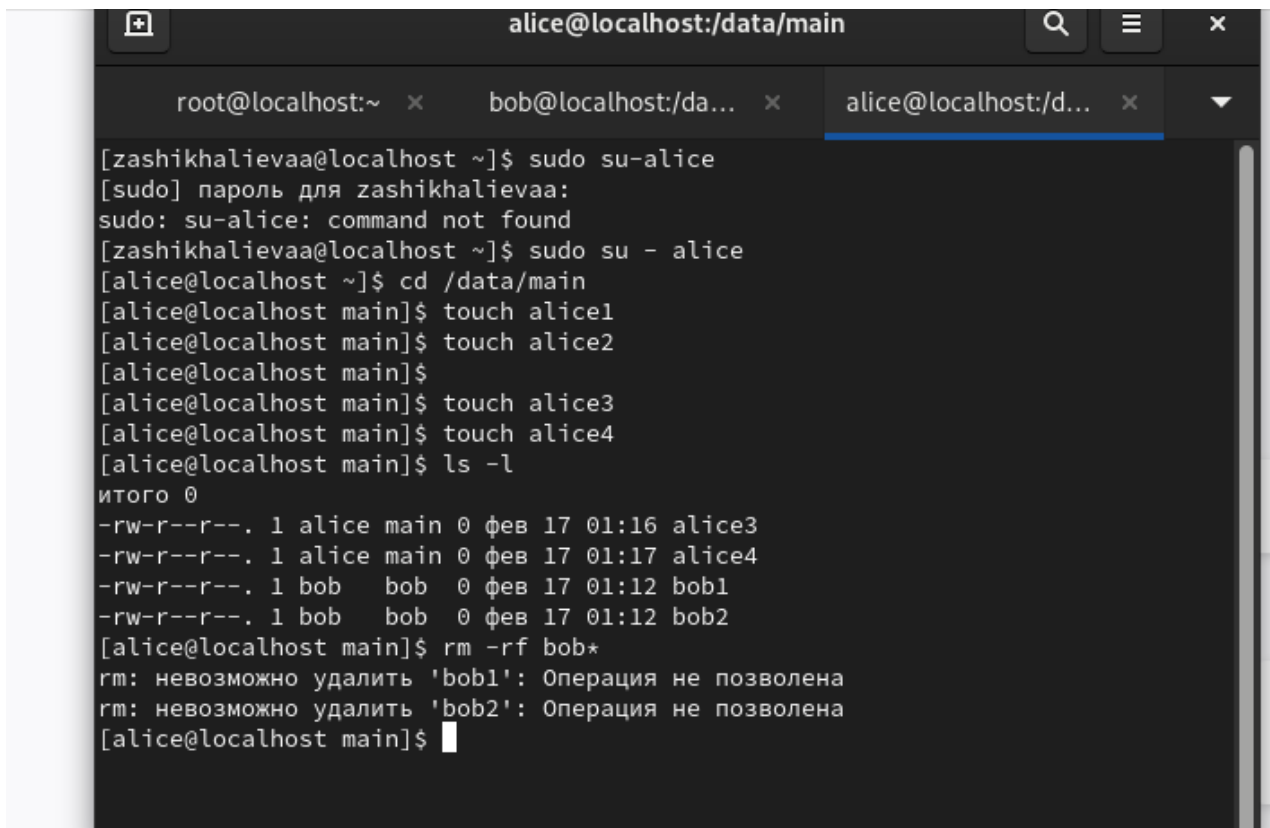
7. В терминале под пользователем alice создайте в каталоге /data/main файлы alice3 и alice4:

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

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

8. В терминале под пользователем alice попробуйте удалить файлы, принадлежащие пользователю bob:

```
rm -rf bob*
```



```
alice@localhost:/data/main

root@localhost:~ x bob@localhost:/da... x alice@localhost:/d... x

[zashikhalievaa@localhost ~]$ sudo su-alice
[sudo] пароль для zashikhalievaa:
sudo: su-alice: command not found
[zashikhalievaa@localhost ~]$ sudo su - alice
[alice@localhost ~]$ cd /data/main
[alice@localhost main]$ touch alice1
[alice@localhost main]$ touch alice2
[alice@localhost 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 фев 17 01:16 alice3
-rw-r--r--. 1 alice main 0 фев 17 01:17 alice4
-rw-r--r--. 1 bob bob 0 фев 17 01:12 bob1
-rw-r--r--. 1 bob bob 0 фев 17 01:12 bob2
[alice@localhost main]$ rm -rf bob*
rm: невозможно удалить 'bob1': Операция не позволена
rm: невозможно удалить 'bob2': Операция не позволена
[alice@localhost main]$
```

Управление расширенными разрешениями с использованием списков ACL

1. Откройте терминал с учётной записью root
su –

2. Установите 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

3. Используйте команду getfacl, чтобы убедиться в правильности установки разрешений:
getfacl /data/main
getfacl /data/third

```
[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:---

[root@localhost ~]#
```

4. Создайте новый файл с именем newfile1 в каталоге /data/main:

```
touch /data/main/newfile1
```

Используйте

```
getfacl /data/main/newfile1
```

для проверки текущих назначений полномочий. Какие права доступа у этого файла?

Объясните, почему.

Выполните аналогичные действия для каталога /data/third. Дайте пояснения.

В каталоге /data/main установлен дополнительный бит g+s (setgid). Это означает, что все файлы и подкаталоги, создаваемые внутри этого каталога, будут автоматически наследовать группу, которой принадлежит сам каталог — в данном случае группу main. Благодаря этому члены группы main, смогут получать доступ к вновь созданным файлам в каталоге, не требуя ручной смены группы для каждого файла. Этот механизм обеспечивает удобное распределение прав доступа между пользователями одной группы, гарантируя, что все новые файлы и каталоги будут принадлежать нужной группе.

На каталоге /data/third бит g+s не установлен. Это значит, что новые файлы, созданные в этом каталоге, будут наследовать группу пользователя, который их создает, а не группу каталога. В результате управление доступом к файлам в этом каталоге будет менее автоматизировано, так как новые файлы могут принадлежать разным группам, в зависимости от того, кто их создавал.

```
mask::rwX
other::---
```



```
[root@localhost ~]#
[root@localhost ~]# touch /data/main/newfile1
[root@localhost ~]# getfacl /data/main/newfile1
getfacl: /data/main/newfile1: Нет такого файла или каталога
[root@localhost ~]# touch /data/main/newfile1
[root@localhost ~]# getfacl /data/main/newfile1
getfacl: /data/main/newfile1: Нет такого файла или каталога
[root@localhost ~]#
```

5. Установите ACL по умолчанию для каталога /data/main:

```
setfacl -m d:g:third:rwX /data/main
```

6. Добавьте ACL по умолчанию для каталога /data/third:

```
setfacl -m d:g:main:rwX /data/third
```

```
[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::---
```



```
[root@localhost ~]#
[root@localhost ~]# touch /data/main/newfile1
[root@localhost ~]# getfacl /data/main/newfile1
getfacl: /data/main/newfile1: Нет такого файла или каталога
[root@localhost ~]# touch /data/main/newfiles1
[root@localhost ~]# getfacl /data/main/newfile1
getfacl: /data/main/newfile1: Нет такого файла или каталога
[root@localhost ~]# setfacl -m d:g:third:rwX /data/main
[root@localhost ~]# setfacl -m d:g:main:rwX /data/third
[root@localhost ~]#
[root@localhost ~]# touch /data/main/newfile2
[root@localhost ~]# getfacl /date/main/newfile2
getfacl: /date/main/newfile2: Нет такого файла или каталога
[root@localhost ~]#
```

7. Убедитесь, что настройки ACL работают, добавив новый файл в каталог /data/main:

```
touch /data/main/newfile2
```

Используйте `getfacl /data/main/newfile2` для проверки текущих назначений полномочий. Выполните аналогичные действия для каталога /data/third.

Используя в команде `setfacl` литеру `d` мы указали применять ACL к директории и к содержимому.

8. Для проверки полномочий группы `third` в каталоге `/data/third` войдите в другом терминале под учётной записью члена группы `third`:

`su - carol`

Проверьте операции с файлами:

`rm /data/main/newfile1`

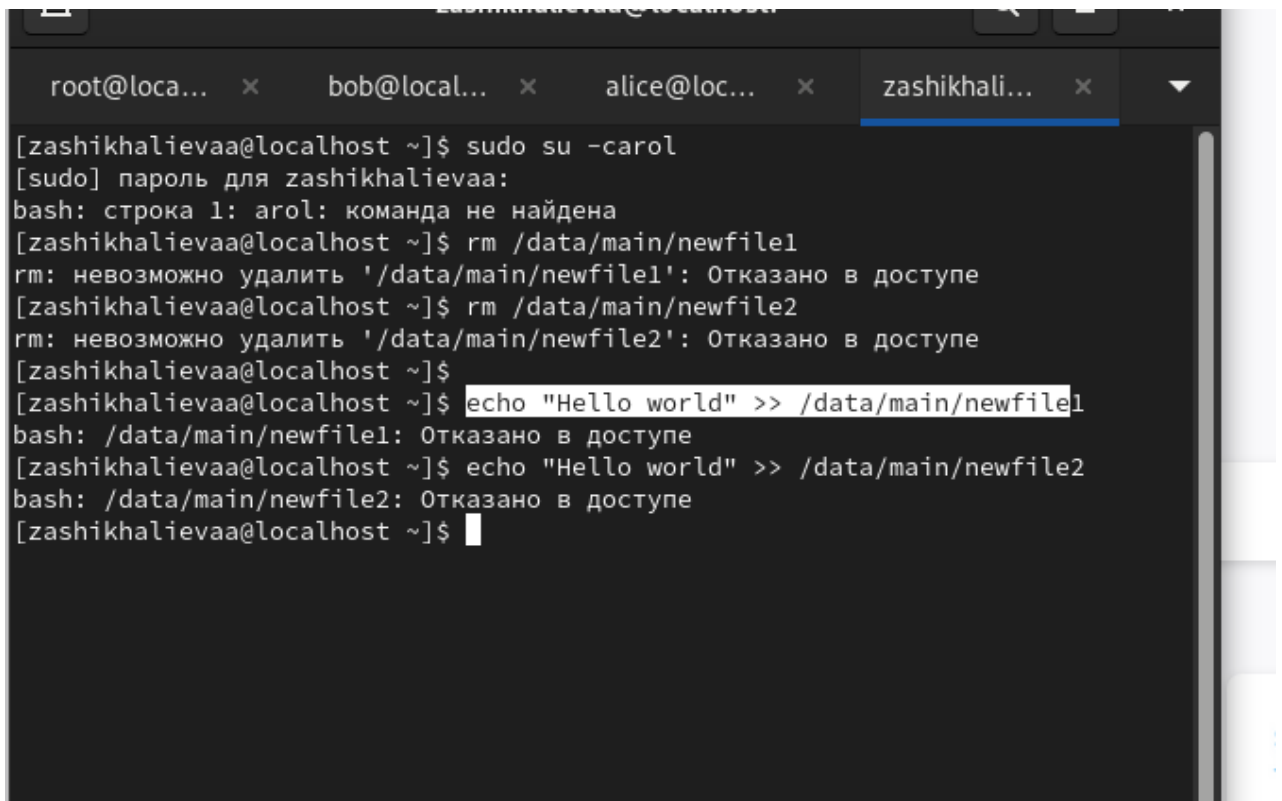
`rm /data/main/newfile2`

Проверьте, возможно ли осуществить запись в файл:

`echo "Hello, world" >> /data/main/newfile1`

`echo "Hello, world" >> /data/main/newfile2`

Объясните результат произведённых действий.



```
root@loca... x bob@local... x alice@loc... x zashikhali... x
[zashikhaliievaa@localhost ~]$ sudo su -carol
[sudo] пароль для zashikhaliievaa:
bash: строка 1: arol: команда не найдена
[zashikhaliievaa@localhost ~]$ rm /data/main/newfile1
rm: невозможно удалить '/data/main/newfile1': Отказано в доступе
[zashikhaliievaa@localhost ~]$ rm /data/main/newfile2
rm: невозможно удалить '/data/main/newfile2': Отказано в доступе
[zashikhaliievaa@localhost ~]$
[zashikhaliievaa@localhost ~]$ echo "Hello world" >> /data/main/newfile1
bash: /data/main/newfile1: Отказано в доступе
[zashikhaliievaa@localhost ~]$ echo "Hello world" >> /data/main/newfile2
bash: /data/main/newfile2: Отказано в доступе
[zashikhaliievaa@localhost ~]$
```

Контрольные вопросы

1. Как следует использовать команду `chown`, чтобы установить владельца группы для файла? Приведите пример.

Для изменения группы-владельца файла используется команда `chown`. Синтаксис следующий: `chown владелец:группа файл`. В этом случае владелец — это новый пользователь, которому будет принадлежать файл, группа — это новая группа, а файл — это имя файла, к которому применяются изменения.

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

Чтобы найти все файлы, принадлежащие определённому пользователю, используется команда `find` с параметром `-user`. Например, для поиска файлов пользователя Alice: `find /home -user alice`.

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.

Для установки прав на чтение, запись и выполнение для всех файлов в каталоге /data для владельцев и их групп, но без прав для других, применяется команда `chmod`. Пример команды: `chmod -R u+rwX,g+rwX /data`.

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

Чтобы сделать файл исполняемым, добавив соответствующее разрешение, используется команда `chmod` с флагом `+x`.

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

Для того чтобы все новые файлы, создаваемые в каталоге, наследовали групповые разрешения этого каталога, нужно установить специальный бит с помощью команды `chmod` с флагом `g+s`. Пример: `chmod g+s 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? Приведите пример.

Чтобы пользователи могли удалять только свои файлы или файлы, находящиеся в каталогах, владельцами которых они являются, используется команда `chmod` с опцией `+t`. Это устанавливает "sticky-бит", который ограничивает удаление файлов только владельцами или администраторами. Пример: `chmod +t directory`.

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

Для добавления ACL, предоставляющего членам группы права на чтение всех существующих файлов в текущем каталоге, используется команда `setfacl`. Пример: `setfacl -m g:groupname:r`.

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? Приведите пример.

Чтобы члены группы могли читать все существующие и будущие файлы в текущем каталоге и его подкаталогах, нужно использовать команду `chmod` с флагом `-R` и установить бит `setgid`. Пример:

```
chmod -R g+rx directory
```

```
chmod g+s directory
```

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

Чтобы запретить другим пользователям доступ к новым файлам, можно установить значение `umask` в `007`. Это гарантирует, что новые файлы будут создаваться с правами `770`. Пример: `umask 007`.

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

Для того чтобы предотвратить случайное удаление файла, можно применить команду `chattr` с опцией `+i`. Это сделает файл неизменяемым. Пример: `chattr +i myfile`.

Заключение

Получены навыки работы с настройкой прав доступа.