Лабораторная работа №3

Установка и конфигурация операционной системы на виртуальную машину

Юсупова Ксения Равилевна

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 Задание

1. Прочитайте справочное описание man по командам chgrp, chmod, getfacl, setfacl.
2. Выполните действия по управлению базовыми разрешениями для групп пользовате- лей (раздел 3.3.1).
3. Выполните действия по управлению специальными разрешениями для групп пользо- вателей (раздел 3.3.2).
4. Выполните действия по управлению расширенными разрешениями с использованием списков ACL для групп пользователей (раздел 3.3.3).

3 Выполнение лабораторной работы

Требуется создать структуру каталогов с разными разрешениями доступа для разных групп пользователей. Сначала открыли терминал с учётной записью root; затем в корневом каталоге создайте каталоги /data/main и /data/third посмотриели, кто является владельцем этих каталогов. Прежде чем устанавливать разрешения, изменили владельцев этих каталогов с root на main и third соответственно. Посмотрели, кто теперь является владельцем этих каталогов. Затем установили разрешения, позволяющие владельцам каталогов записывать файлы в эти каталоги и запрещающие доступ к содержимому каталогов всем другим пользователям и группам.(рис. 3.1).

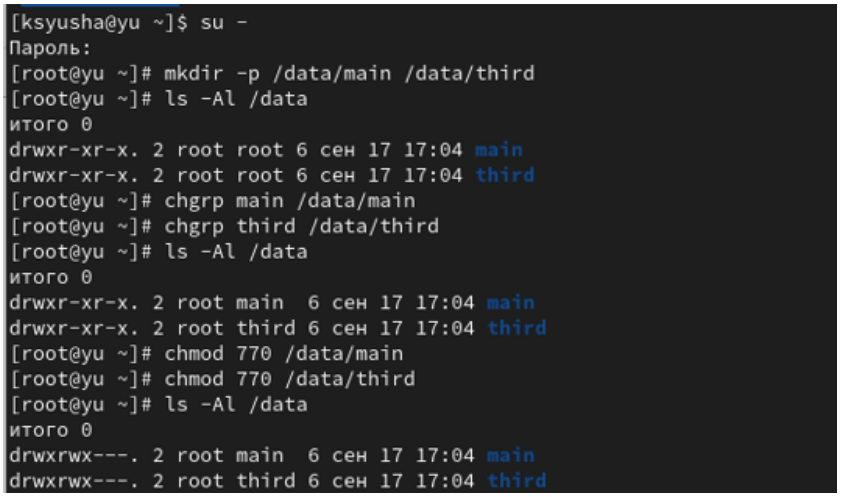

Рис. 3.1: Выполнили пункты 1-4 из раздела 3.3.1 (Управление базовыми разрешениями)

В другом терминале перешли под учётную запись пользователя bob и перешли в каталог /data/main, создали файл emptyfile. В этом каталоге получилось успешно

создать файл emptyfile создан, так как bob имеет права на выполнение (x) и запись (w) в каталоге. В каталоге /data/third было отказано в доступе, так как bob не имеет права на выполнение (x) для входа в этот каталог.(рис. 3.2).

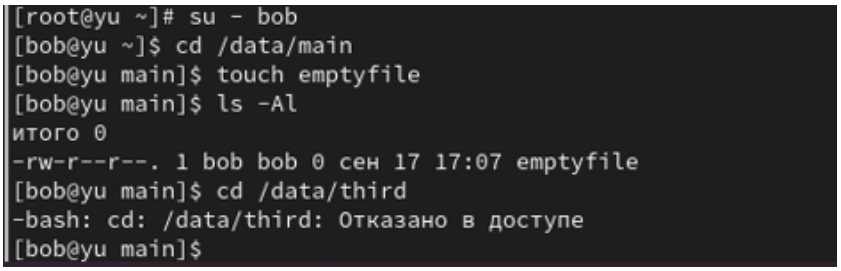

Рис. 3.2: Выполнили пункты 5-7 из раздела 3.3.1 (Управление базовыми разрешениями)

Открыли новый терминал под пользователем alice и перешли в каталог /data/main. Создали два файла, владельцем которых является alice: touch alice1, touch alice2 (рис. 3.3).

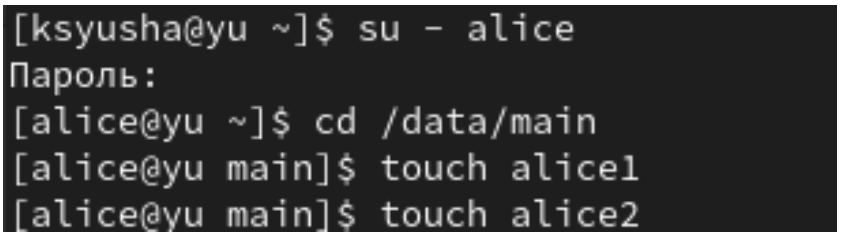

Рис. 3.3: Выполнили пункты 1 и 2 из раздела 3.3.2 (Управление специальными разрешениями)

В другом терминале перешли под учётную запись пользователя bob и в каталог /data/main. Увидили два файла, созданные пользователем alice. Удалили файлы, принадлежащие пользователю alice; убедились что файлы будут удалены пользователем bob. Затем создали два файла, которые принадлежат пользователю bob: touch bob1, touch bob2(рис. 3.4).

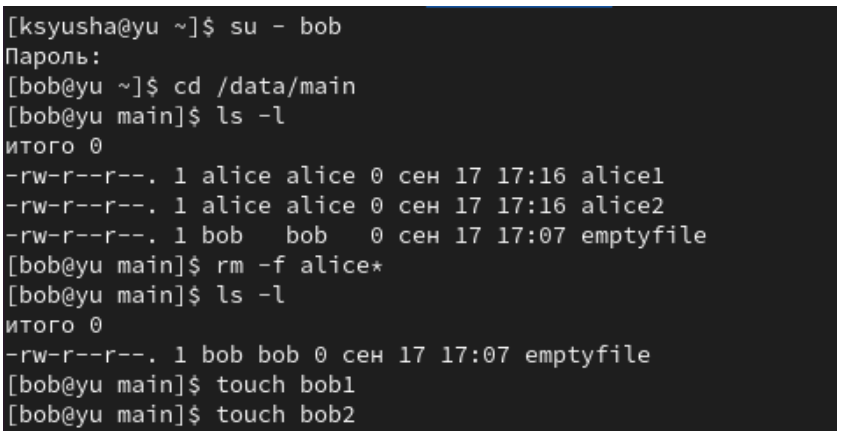

Рис. 3.4: Выполнили пункты 3-5 из раздела 3.3.2 (Управление специальными разрешениями)

В терминале под пользователем root установили для каталога /data/main бит идентификатора группы, а также stiky-бит для разделяемого (общего) каталога группы:(рис. 3.5).

Рис. 3.5: Выполнили пункт 6 из раздела 3.3.2 (Управление специальными разрешениями)

В терминале под пользователем alice создали в каталоге /data/main файлы alice3 и alice4; затем увидели, что два созданных вами файла принадлежат группе main, которая является группой-владельцем каталога /data/main. В терминале под пользователем alice попробовали удалить файлы, принадлежащие пользователю bob; и убедились, что sticky-bit предотвратит удаление этих файлов пользователем alice, поскольку этот пользователь не является владельцем этих файлов. Обратили внимание: поскольку пользователь alice является владельцем каталога /data/main, то он может удалить все свои файлы в любом случае. ([рис. **fig?**]).

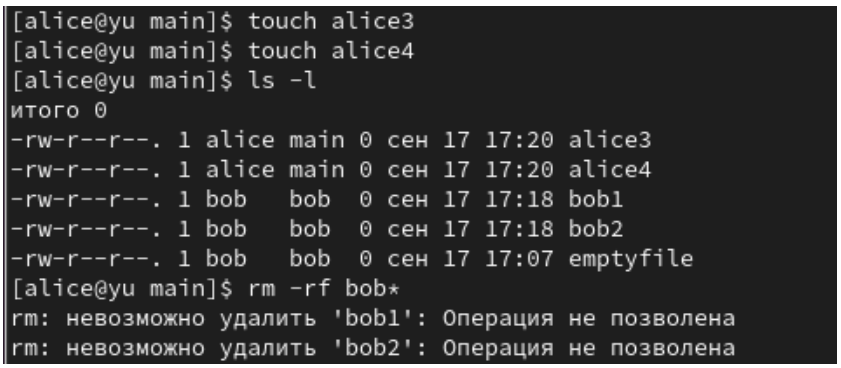

Рис. 3.6: Выполнили пункты 7 и 8 из раздела 3.3.2 (Управление специальными разрешениями)

Открыли терминал с учётной записью root; установили 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. Затем использовали команду getfacl, чтобы убедиться в правильности установки разрешений(рис. 3.7).

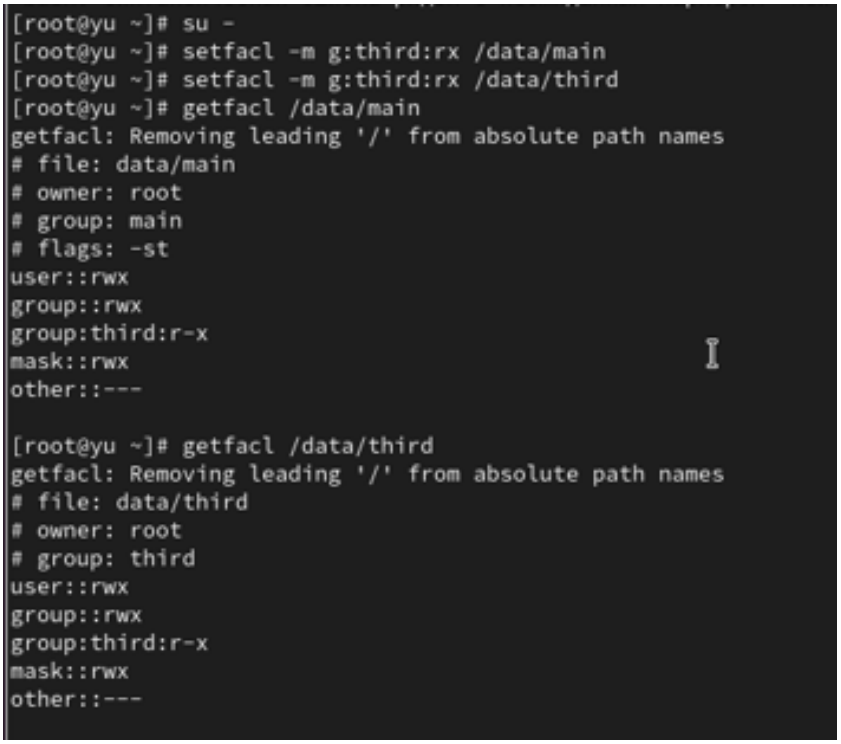

Рис. 3.7: Выполнили пункты 1-3 из раздела 3.3.3 (Управление расширенными разрешениями с использованием списков ACL)

Создали новый файл с именем newfile1 в каталоге /data/main, затем исполь-

зовали проверку текущих назначений полномочий. Проверка прав показывает следующие права доступа: владелец файла - root с правами чтение и запись (rw-), группа-владелец - main с правами чтение и запись (rw-), другие пользователи не имеют никаких прав (—). Такие права унаследованы потому, что файл создается с учетом базовых прав каталога и установленного umask. Группа-владелец файла соответствует группе-владельцу каталога благодаря ранее установленному SGID-биту (set group ID).

Для каталога /data/third аналогичные действия выполнить не удается, так как пользователь не имеет прав доступа к этому каталогу - команда cd /data/third возвращает ошибку “Отказано в доступе”. Это связано с отсутствием права на выполнение (x) для данного пользователя на каталог /data/third

Установили ACL по умолчанию для каталога /data/main и /data/third. Убедились, что настройки ACL работают, добавив новый файл в каталог /data/main. Проверили текущее назначение полномочий. И выполнили аналогичные действия для каталога /data/third (рис. 3.8).

```
[root@yu ~]# touch /data/main/newfile1
[root@yu ~]# 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@yu ~]# setfacl -m d:g:third:rwx /data/main
[root@yu ~]# setfacl -m d:g:main:rwx /data/third
[root@yu ~]# touch /data/main/newfile2
[root@yu ~]# 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::---
```

Рис. 3.8: Выполнили пункты 4-7 из раздела 3.3.3 (Управление расширенными разрешениями с использованием списков ACL)

Для проверки полномочий группы third в каталоге /data/third воши в другом терминале под учётной записью члена группы third, и проверили операции с файлами: rm /data/main/newfile1 и rm /data/main/newfile2. Причина невозможности удаления: Для удаления файла необходимо право записи (w) не на сам файл, а на родительский каталог, где находится файл. Пользователь carol не имеет права записи в каталог /data/main. Проверили, возможно ли осуществить запись в файлы. ACL (Access Control Lists) позволяют гибко управлять правами доступа на уровне отдельных файлов/каталогов, предоставляя права конкретным пользователям или группам поверх базовых прав UNIX. В данном случае: базовые права запрещали запись для всех, кроме владельца и группы main; а ACL предоставил право записи группе third только для новых файлов (созданных после установки ACL)(рис. 3.9).

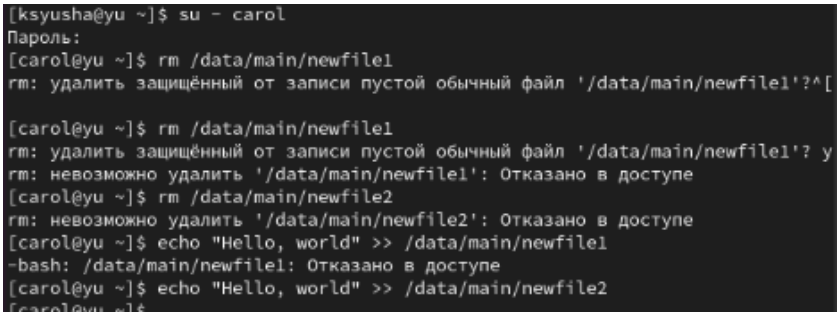

Рис. 3.9: Выполнили пункт 8 из раздела 3.3.3 (Управление расширенными разрешениями с использованием списков ACL)

4 Ответы на контрольные вопросы

1. Чтобы установить владельца группы для файла с помощью команды chown, используется синтаксис `chown :GROUPNAME FILE`. Например, команда `chown :developers script.sh` установит группу ‘developers’ в качестве владельца для файла script.sh. Для рекурсивного изменения группы всего каталога используется флаг -R: `chown -R :www-data /var/www/html`.

2. Для поиска всех файлов, принадлежащих конкретному пользователю, используется команда find с опцией -user: `find PATH -user USERNAME -type f`. Например, `find /home -user bob -type f` найдет все файлы пользователя bob в домашней директории. Для поиска по всей системе с игнорированием ошибок доступа используется `find / -user alice -type f 2>/dev/null`.

3. Чтобы применить разрешения на чтение, запись и выполнение для всех файлов в каталоге /data для пользователей и владельцев групп без прав для других, используется команда `chmod -R ug=rwX,o= /data`. Флаг -R обеспечивает рекурсивное применение, ug=rwX устанавливает права чтения и записи для пользователя и группы с выполнением только для каталогов (X), а o= удаляет все права для других пользователей.

4. Для добавления разрешения на выполнение для файла используется команда `chmod +x FILENAME`. Например, `chmod +x script.sh` сделает файл исполняемым для всех пользователей. Для установки права выполнения только для пользователя и группы используется `chmod ug+x script.sh`.

5. Чтобы гарантировать, что групповые разрешения для всех новых файлов будут присвоены владельцу группы каталога, используется команда `chmod g+s`

DIRECTORY. Например, chmod g+s /shared установит SGID бит, при котором новые файлы будут наследовать группу-владельца каталога /shared.

6. Для обеспечения возможности удаления только собственных файлов используется команда chmod +t DIRECTORY, которая устанавливает sticky bit. Например, chmod +t /shared/tmp разрешит удаление файлов в каталоге /shared/tmp только их владельцам, даже если каталог имеет широкие права доступа.

7. Для добавления ACL, предоставляющего членам группы права доступа на чтение для всех существующих файлов в текущем каталоге, используется команда setfacl -R -m g:GROUPNAME:r .. Например, setfacl -R -m g:readers:r . предоставит группе readers права на чтение всех файлов в текущем каталоге рекурсивно.

8. Для гарантии прав на чтение для членов группы для всех текущих и будущих файлов используется команда setfacl -R -d -m g:GROUPNAME:r DIRECTORY. Например, setfacl -R -d -m g:readers:r . установит права по умолчанию для группы readers, а комбинация с setfacl -R -m g:readers:r . обеспечит права для существующих файлов.

9. Чтобы другие пользователи не получали какие-либо разрешения на новые файлы, устанавливается umask 007. Например, после выполнения umask 007 создаваемые файлы будут иметь права 660 (rw-rw—-), что дает полные права пользователю и группе без прав для других.

10. Для защиты файла от случайного удаления используется команда chattr +i myfile, которая устанавливает атрибут “immutable”. Например, chattr +i important_file.txt сделает файл неизменяемым - его нельзя будет удалить, переименовать или изменить. Снять защиту можно командой chattr -i important_file.txt.

5 Выводы

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