

Отчёт по лабораторной работе №3

Основы информационной безопасности

Надежда Александровна Рогожина

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	8
4	Выполнение лабораторной работы	11
5	Выводы	23
	Список литературы	24

Список иллюстраций

4.1	Добавление нового пользователя	11
4.2	Добавление нового пользователя в группу	11
4.3	Добавление нового пользователя	12
4.4	guest2:pwd	12
4.5	guest:pwd	12
4.6	narogozhina:pwd	12
4.7	groups guest	13
4.8	groups guest2	13
4.9	groups id	13
4.10	groups id	14
4.11	id	14
4.12	newgrp guest	14
4.13	chmod	15
4.14	chmod check	15
4.15	chmod 000	15
4.16	d---	16
4.17	d--x-	16
4.18	d--x-	17
4.19	d-r--	17
4.20	d-r--	18
4.21	d--w--	18
4.22	d--w--	18
4.23	d-r-x-	19
4.24	d-r-x-	19
4.25	d-rw--	19
4.26	d-rw--	20
4.27	d--wx-	20
4.28	d--wx-	21

Список таблиц

3.1	Описание некоторых команд Linux	10
4.1	Установленные права и разрешённые действия	21

1 Цель работы

Получение практических навыков работы в консоли с атрибутами файлов, закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux.

2 Задание

1. В установленной при выполнении предыдущей лабораторной работы операционной системе создайте учётную запись пользователя `guest` (используя учётную запись администратора): `useradd guest`
2. Задайте пароль для пользователя `guest` (используя учётную запись администратора): `passwd guest`
3. Аналогично создайте второго пользователя `guest2`.
4. Добавьте пользователя `guest2` в группу `guest`: `gpasswd -a guest2 guest`
5. Осуществите вход в систему от двух пользователей на двух разных консолях: `guest` на первой консоли и `guest2` на второй консоли.
6. Для обоих пользователей командой `pwd` определите директорию, в которой вы находитесь. Сравните её с приглашениями командной строки.
7. Уточните имя вашего пользователя, его группу, кто входит в неё и к каким группам принадлежит он сам. Определите командами `groups guest` и `groups guest2`, в какие группы входят пользователи `guest` и `guest2`. Сравните вывод команды `groups` с выводом команд `id -Gn` и `id -G`.
8. Сравните полученную информацию с содержимым файла `/etc/group`. Просмотрите файл командой `cat /etc/group`.
9. От имени пользователя `guest2` выполните регистрацию пользователя `guest2` в группе `guest` командой `newgrp guest`.

10. От имени пользователя guest измените права директории /home/guest, разрешив все действия для пользователей группы: `chmod g+rx /home/guest`
11. От имени пользователя guest снимите с директории /home/guest/dir1 все атрибуты командой `chmod 000 dir1` и проверьте правильность снятия атрибутов. Меняя атрибуты у директории dir1 и файла file1 от имени пользователя guest и делая проверку от пользователя guest2, заполните табл. 3.1, определив опытным путём, какие операции разрешены, а какие нет. Если операция разрешена, занесите в таблицу знак «+», если не разрешена, знак «-».

Сравните табл. 2.1 (из лабораторной работы № 2) и табл. 3.1. На основании заполненной таблицы определите те или иные минимально необходимые права для выполнения пользователем guest2 операций внутри директории dir1 и заполните табл. 3.2

3 Теоретическое введение

Есть 3 вида разрешений. Они определяют права пользователя на 3 действия: чтение, запись и выполнение. В Linux эти действия обозначаются вот так:

- r — read (чтение) — право просматривать содержимое файла;
- w — write (запись) — право изменять содержимое файла;
- x — execute (выполнение) — право запускать файл, если это программа или скрипт.

У каждого файла есть 3 группы пользователей, для которых можно устанавливать права доступа.

- owner (владелец) — отдельный человек, который владеет файлом. Обычно это тот, кто создал файл, но владельцем можно сделать и кого-то другого.
- group (группа) — пользователи с общими заданными правами.
- others (другие) — все остальные пользователи, не относящиеся к группе и не являющиеся владельцами.

Как узнать разрешения файла?

Чтобы посмотреть права доступа к файлу, нужно вызвать команду `ls` с опцией `-l`. Эта опция отвечает за вывода списка в длинном формате.

```
ls -l <путь>
```

```
user@bash: ls -l /home/karpaff/linuxtutorialwork/chick.png
```

```
-rwxr---x 1 harry users 2.7K Jan 4 07:32 /home/karpaff/linuxtutorialwork/chick.p
```

```
user@bash:
```


В приведенном выше примере первые 10 символов вывода показывают разрешения файла.

- Первый символ определяет тип файла. Если первый символ — прочерк -, то это обычный файл. Если первый символ d, то это каталог.
- Следующие 3 символа показывают разрешения для владельца. Буква означает наличие разрешения, а прочерк - — его отсутствие. В нашем примере у владельца есть все разрешения (чтение, запись и выполнение).
- Следующие 3 символа показывают разрешения для группы. В этом примере у членов группы есть разрешение на чтение, но нет разрешений на запись и выполнение. Обратите внимание, порядок записи разрешений всегда такой: чтение, запись, выполнение.
- Последние 3 символа показывают разрешения для всех остальных пользователей. В этом примере у них есть только разрешение на выполнение.

Как изменить права доступа?

Для изменения прав доступа к файлу или каталога используется команда `chmod` (от англ. change mode). Эта команда меняет биты режима файла — если совсем просто, это индикатор разрешений.

`chmod [разрешение] [путь]`

Аргументы команды `chmod`, отвечающие за разрешение, состоят из 3 компонентов:

- Для кого мы меняем разрешение? Обозначается первыми буквами слов: `[ugoa]` — user (пользователь, он же владелец), group (группа), others (другие), all (все).
- Мы предоставляем или отзываем разрешения? Обозначается плюсом +, если предоставляем, минусом -, если отзываем.
- Какое разрешение мы хотим изменить? Чтение (r), запись (w), выполнение (x).

В табл. [3.1] приведено краткое описание основных команд, использованных в лабораторной работе.

Таблица 3.1: Описание некоторых команд Linux

Имя каталога	Описание каталога
cd	Смена директории
clear	Очищение окна терминала
id	Отображение системного идентификатора указанного пользователя
whoami	Имя пользователя
group	Группа, к которой принадлежит пользователь
sudo	Права суперпользователя
cat	Отражение содержимого текстового файла в терминале
chmod	Изменение прав доступа к файлу
ls	Просмотр содержимого директории
touch	Создание файла
mkdir	Создание директории
rm	Удаление файла
mv	Перемещение/переименование файла

4 Выполнение лабораторной работы

В первую очередь добавить нового пользователя и установим для него пароль (рис.[4.1]):

```
[narogozhina@narogozhina ~]$ useradd guest2
useradd: Permission denied.
useradd: cannot lock /etc/passwd; try again later.
[narogozhina@narogozhina ~]$ sudo -i
[sudo] password for narogozhina:
[root@narogozhina ~]# useradd guest2
[root@narogozhina ~]# passwd guest2
Changing password for user guest2.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
[root@narogozhina ~]# passwd guest2
Changing password for user guest2.
New password:
BAD PASSWORD: The password fails the dictionary check - it is based on a dictionary word
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
[root@narogozhina ~]# passwd guest2
Changing password for user guest2.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@narogozhina ~]#
```

Рис. 4.1: Добавление нового пользователя

Добавляем guest2 в группу guest (рис.[4.2]):

```
[root@narogozhina ~]# gpasswd -a guest2 guest
Adding user guest2 to group guest
[root@narogozhina ~]#
```

Рис. 4.2: Добавление нового пользователя в группу

С основной консоли войдем в основной аккаунт, а с двух других - в guest и guest2 (рис.[4.3]):

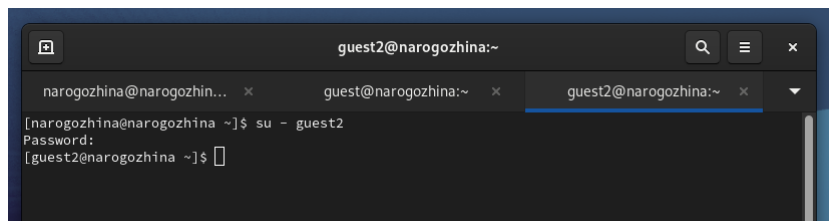


Рис. 4.3: Добавление нового пользователя

Проверка путей (рис.[4.4], [4.5]):

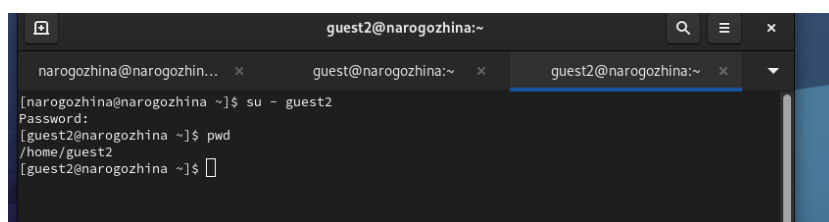


Рис. 4.4: guest2:pwd

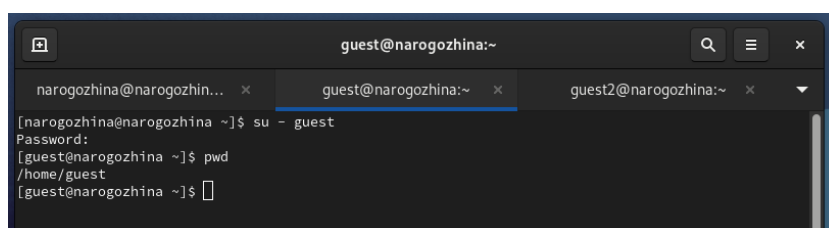


Рис. 4.5: guest:pwd

Путь домашнего каталога от основного пользователя (рис.[4.6]):

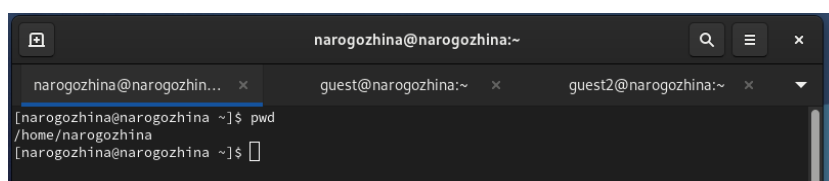


Рис. 4.6: narogozhina:pwd

Groups guest (рис.[4.7]):

```
[guest@narogozhina ~]$ groups guest
guest : guest
[guest@narogozhina ~]$
```

Рис. 4.7: groups guest

Groups guest2 (рис.[4.8]):

```
/home/guest
[guest@narogozhina ~]$ groups guest
guest : guest
[guest@narogozhina ~]$ groups guest2
guest2 : guest2 guest
[guest@narogozhina ~]$
```

Рис. 4.8: groups guest2

groups VS id -G for guest (рис.[4.9]):

```
[guest@narogozhina ~]$ groups guest
guest : guest
[guest@narogozhina ~]$ groups guest2
guest2 : guest2 guest
[guest@narogozhina ~]$ id -Gn
guest
[guest@narogozhina ~]$ id -G
1001
[guest@narogozhina ~]$
```

Рис. 4.9: groups|id

groups VS id -G for guest2 (рис.[4.10], [4.11]):

```
[guest2@narogozhina ~]$ groups guest
guest : guest
[guest2@narogozhina ~]$ groups guest2
guest2 : guest2 guest
[guest2@narogozhina ~]$ id -Gn
guest2 guest
[guest2@narogozhina ~]$ id -G
1002 1001
[guest2@narogozhina ~]$
```

Рис. 4.10: groupslid

```
vboxdrmipc:x:975:
guest:x:1001:guest2
guest2:x:1002:
[guest@narogozhina ~]$
```

Рис. 4.11: id

Добавление guest2 в группу guest (рис.[4.12]):

```
guest2:x:1002:
[guest2@narogozhina ~]$ newgrp guest
[guest2@narogozhina ~]$
```

Рис. 4.12: newgrp guest

От имени guest2 изменим права доступа группы (рис.[4.13]):

```

[guest@narogozhina ~]$ chmod g+rx /home/guest
[guest@narogozhina ~]$

```

Рис. 4.13: chmod

Проверка (рис.[4.14]):

```

guest2:x:1002:
[guest2@narogozhina ~]$ newgrp guest
[guest2@narogozhina ~]$ cd /home/guest
[guest2@narogozhina guest]$

```

Рис. 4.14: chmod check

Все получилось. Далее убираем все гостевые права с этой директории (рис.[4.15]):

```

[guest@narogozhina ~]$ chmod g+rx /home/guest
[guest@narogozhina ~]$ chmod 000 dir1
[guest@narogozhina ~]$ ls
dir1
[guest@narogozhina ~]$ ls -l
total 0
d----- . 2 guest guest 6 Mar 1 14:30 dir1
[guest@narogozhina ~]$

```

Рис. 4.15: chmod 000

Далее я приведу скриншоты, по которым я заполняла таблицу 2.1 (рис. [4.16], рис. [4.17], рис. [4.18], рис.[4.19], рис.[4.20], рис.[4.21], рис.[4.22], рис.[4.23], рис.[4.24], рис.[4.25], рис.[4.26], рис.[4.27], рис.[4.28]):

```

[guest2@narogozhina ~]$ newgrp guest
[guest2@narogozhina ~]$ cd /home/guest
[guest2@narogozhina guest]$ cd ~
[guest2@narogozhina ~]$ cd /home/guest
[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
bash: cd: dir1: Permission denied
[guest2@narogozhina guest]$

```

Рис. 4.16: d---—

```

[guest@narogozhina ~]$ ls -l
total 0
d----- . 2 guest guest 6 Mar  1 14:30 dir1
[guest@narogozhina ~]$ chmod g+x /home/guest/dir1
[guest@narogozhina ~]$ ls -l
total 0
d-----x . 2 guest guest 6 Mar  1 14:30 dir1
[guest@narogozhina ~]$

```

Рис. 4.17: d---x—


```

[guest2@narogozhina dir1]$ ls
ls: cannot open directory '.': Permission denied
[guest2@narogozhina dir1]$ cd ~
[guest2@narogozhina ~]$ ls
[guest2@narogozhina ~]$ cd dir1
bash: cd: dir1: No such file or directory
[guest2@narogozhina ~]$ cd /home/guest
[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
[guest2@narogozhina dir1]$ ls
ls: cannot open directory '.': Permission denied
[guest2@narogozhina dir1]$ chmod g+r file1
[guest2@narogozhina dir1]$ touch file2
touch: cannot touch 'file2': Permission denied
[guest2@narogozhina dir1]$ rm file1
rm: cannot remove 'file1': Permission denied
[guest2@narogozhina dir1]$ echo 'hello'>>file1
[guest2@narogozhina dir1]$ cat file1
hello
[guest2@narogozhina dir1]$ chmod g-r file1
[guest2@narogozhina dir1]$ cat file1
hello
[guest2@narogozhina dir1]$ ls -l
ls: cannot open directory '.': Permission denied
[guest2@narogozhina dir1]$

```

Рис. 4.18: d--x--

```

[guest@narogozhina ~]$ chmod g-x /home/guest/dir1
[guest@narogozhina ~]$ chmod g+r /home/guest/dir1
[guest@narogozhina ~]$

```

Рис. 4.19: d-r--

```

[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
bash: cd: dir1: Permission denied
[guest2@narogozhina guest]$ ls dir1
ls: cannot access 'dir1/file1': Permission denied
file1
[guest2@narogozhina guest]$ touch /home/guest/dir1/ file1
touch: setting times of '/home/guest/dir1/': Permission denied
[guest2@narogozhina guest]$ rm file1
[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ rm /home/guest/dir1/file1
rm: cannot remove '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$ echo 'hi' >> /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$ cat /home/guest/dir1/file1
cat: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$ mv /home/guest/dir1/file1 /home/guest/dir1/fil2
mv: failed to access '/home/guest/dir1/fil2': Permission denied
[guest2@narogozhina guest]$ chmod g+wx /home/guest/dir1/file1
chmod: cannot access '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$

```

Рис. 4.20: d—r—

```

[guest@narogozhina ~]$ chmod g-r /home/guest/dir1
[guest@narogozhina ~]$ chmod g+w /home/guest/dir1
[guest@narogozhina ~]$

```

Рис. 4.21: d—-w—-

```

[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
bash: cd: dir1: Permission denied
[guest2@narogozhina guest]$ chmod g+wx /home/guest/dir1/file1
chmod: cannot access '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$ mv /home/guest/dir1/file1 /home/guest/dir1/file2
mv: failed to access '/home/guest/dir1/file2': Permission denied
[guest2@narogozhina guest]$ touch /home/guest/dir1/file2
touch: cannot touch '/home/guest/dir1/file2': Permission denied
[guest2@narogozhina guest]$ rm /home/guest/dir1/file1
rm: cannot remove '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$ cat /home/guest/dir1/file1
cat: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$ echo 'hi' >> /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$

```

Рис. 4.22: d—-w—-

```
[guest@narogozhina ~]$ chmod g-w /home/guest/dir1
[guest@narogozhina ~]$ chmod g+rx /home/guest/dir1
[guest@narogozhina ~]$
```

Рис. 4.23: d—r-x—

```
[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
[guest2@narogozhina dir1]$ ls
file1
[guest2@narogozhina dir1]$ ls -l
total 4
-rw---r--. 1 guest2 guest 6 Mar 16 20:08 file1
[guest2@narogozhina dir1]$ chmod g+wx file1
[guest2@narogozhina dir1]$ ls -l
total 4
-rw--wxr--. 1 guest2 guest 6 Mar 16 20:08 file1
[guest2@narogozhina dir1]$ chmod g+rw file1
[guest2@narogozhina dir1]$ ls -l
total 4
-rw-rwxr--. 1 guest2 guest 6 Mar 16 20:08 file1
[guest2@narogozhina dir1]$ touch file2
touch: cannot touch 'file2': Permission denied
[guest2@narogozhina dir1]$ echo 'hi'>>file1
[guest2@narogozhina dir1]$ cat file1
hello
hi
[guest2@narogozhina dir1]$ mv file1 file11
mv: cannot move 'file1' to 'file11': Permission denied
[guest2@narogozhina dir1]$ rm file1
rm: cannot remove 'file1': Permission denied
[guest2@narogozhina dir1]$
```

Рис. 4.24: d—r-x—

```
[guest@narogozhina ~]$ chmod g-x /home/guest/dir1
[guest@narogozhina ~]$ chmod g+w /home/guest/dir1
[guest@narogozhina ~]$
```

Рис. 4.25: d—rw—

```

[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
bash: cd: dir1: Permission denied
[guest2@narogozhina guest]$ ls dir1
ls: cannot access 'dir1/file1': Permission denied
file1
[guest2@narogozhina guest]$ chmod g-x /home/guest/dir1/file1
chmod: cannot access '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$ mv /home/guest/dir1/file1 /home/guest/dir1/file11
mv: failed to access '/home/guest/dir1/file11': Permission denied
[guest2@narogozhina guest]$ touch /home/guest/dir1/file2
touch: cannot touch '/home/guest/dir1/file2': Permission denied
[guest2@narogozhina guest]$ echo 'hihi' >> /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$ cat /home/guest/dir1/file1
cat: /home/guest/dir1/file1: Permission denied
[guest2@narogozhina guest]$ rm /home/guest/dir1/file1
rm: cannot remove '/home/guest/dir1/file1': Permission denied
[guest2@narogozhina guest]$ 

```

Рис. 4.26: d—rw—

```

[guest@narogozhina ~]$ chmod g-r /home/guest/dir1
[guest@narogozhina ~]$ chmod g+x /home/guest/dir1
[guest@narogozhina ~]$ ls -l
total 0
d---wx---. 2 guest guest 19 Mar 16 20:04 dir1
[guest@narogozhina ~]$ 

```

Рис. 4.27: d---wx—

```
[guest2@narogozhina guest]$ ls
dir1
[guest2@narogozhina guest]$ cd dir1
[guest2@narogozhina dir1]$ ls
ls: cannot open directory '.': Permission denied
[guest2@narogozhina dir1]$ ls -l
ls: cannot open directory '.': Permission denied
[guest2@narogozhina dir1]$ mv file1 file11
[guest2@narogozhina dir1]$ chmod g-x file11
[guest2@narogozhina dir1]$ chmod g+x file11
[guest2@narogozhina dir1]$ touch file2
[guest2@narogozhina dir1]$ echo 'hi' >> file2
[guest2@narogozhina dir1]$ cat file2
hi
[guest2@narogozhina dir1]$ rm file2
[guest2@narogozhina dir1]$
```

Рис. 4.28: d--wx—

После проведенных тестов, итоговая таблица[4.1] заполнена:

Таблица 4.1: Установленные права и разрешённые действия

Права директо- рии	Пра- ва фай- ла	Со- зда- ние фай- ла	Уда- ле- ние фай- ла	За- пись в файл	Чте- ние фай- ла	Сме- на ди- рек- то- рии	Про- смотр файлов в директо- рии	Пере- име- нова- ние фай- ла	Сме- на ат- ри- бу- тов фай- ла
d(000)	000	-	-	-	-	-	-	-	-
d--	001	-	-	-	-	+	-	-	+
x-----									
dr-----	100	-	-	-	-	-	+	-	-

Права директо- рии	Пра- ва фай- ла	Со- зда- ние фай- ла	Уда- ле- ние фай- ла	За- пись в файл	Чте- ние фай- ла	Сме- на ди- рек- то- рии	Про- смотр файлов в директо- рии	Пере- име- нова- ние фай- ла	Сме- на ат- ри- бу- тов фай- ла
d-	010	+	-	-	-	-	+	-	-
w-----									
drw-----	110	-	-	-	-	-	+	-	-
dr-	101	-	-	+	+	+	+	-	+
x-----									
d-	011	+	+	+	+	+	-	+	+
wx-----									
drwx-----	111	+	+	+	+	+	+	+	+

После создания предыдущей таблицы, можем заполнить минимальные права для совершения операций : |||| | Операция | Минимальные права на директорию | Минимальные права на файл | | Создание файла | x | w | | Удаление файла | x | w | | Чтение файла | r | r | | Запись в файл | wx | w | | Переименование файла | wx | x | | Создание поддиректории | wx | wx | | Удаление поддиректории | wx | wx | ||||

5 Выводы

В ходе выполнения работы мы получили практические навыки работы в консоли с атрибутами файлов и закрепили теоретические основы разграничения доступа в современных системах с открытым кодом на базе ОС Linux.

Список литературы