

# **Лабораторная работа №2**

**Дисциплина: основы информационной безопасности**

Астраханцева А. А.

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Заполнение таблицы 2.1 . . . . .	12
4.2	Заполнение таблицы 2.2 . . . . .	15
<b>5</b>	<b>Выводы</b>	<b>17</b>
<b>6</b>	<b>Список литературы. Библиография</b>	<b>18</b>

## Список иллюстраций

4.1	Создание новой учетной записи и установка пароля . . . . .	8
4.2	Вход в новую учетную запись . . . . .	9
4.3	Перезод в домашний каталог, уточнение имени пользователя . .	9
4.4	Вывод имени пользователя, его группы и т. д. . . . .	10
4.5	Просмотр файла /etc/passwd . . . . .	10
4.6	Просмотр поддиректорий home . . . . .	10
4.7	Просмотр атрибутов . . . . .	11
4.8	Создание новой директории и просмотр прав доступа на нее . .	11
4.9	Снятие всех атрибутов с созданной директории . . . . .	12
4.10	Попытка создать файл в директории со снятыми атрибутами парв доступа . . . . .	12

## **Список таблиц**

# 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. Выполнить все задания из списка
2. Составить 2 таблицы по правам доступа

## 3 Теоретическое введение

Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенными файлами и каталогами. [1] Есть 3 вида разрешений. Они определяют права пользователя на 3 действия: чтение, запись и выполнение. В Linux эти действия обозначаются вот так:

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

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

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

Инструкция по выполнению лабораторной работы была взята с портала ТУИС [2].

## 4 Выполнение лабораторной работы

1. В установленной при выполнении предыдущей лабораторной работы операционной системе создаем учётную запись пользователя guest (рис. 4.1).

```
[root@aaastrakhanceva user]# useradd guest
[root@aaastrakhanceva user]# passwd guest
Changing password for user guest.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
[root@aaastrakhanceva user]# passwd guest
Changing password for user guest.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
Sorry, passwords do not match.
passwd: Authentication token manipulation error
[root@aaastrakhanceva user]# passwd guest
Changing password for user guest.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
[root@aaastrakhanceva user]#
```

Рис. 4.1: Создание новой учетной записи и установка пароля

2. Входим в систему от имени пользователя guest (рис. 4.2).



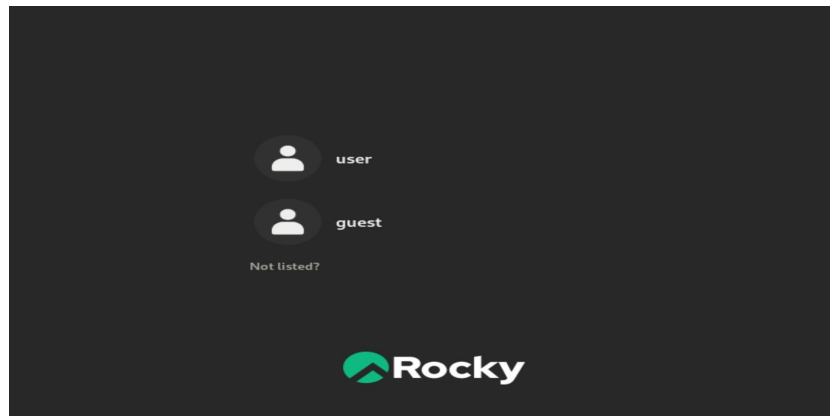


Рис. 4.2: Вход в новую учетную запись

3. Определяем директорию, в которой находимся, командой `pwd`. Сравниваем её с приглашением командной строки. Определяем, является ли она домашней директорией с помощью `cd ~` (переход в корневой каталог), и снова вводим `pwd`? Уточняем имя пользователя командой `whoami` (рис. 4.3).

```
[guest@aaastrakhanceva ~]$ pwd
/home/guest
[guest@aaastrakhanceva ~]$ cd ~
[guest@aaastrakhanceva ~]$ pwd
/home/guest
[guest@aaastrakhanceva ~]$ whoami
guest
[guest@aaastrakhanceva ~]$
```

Рис. 4.3: Перезод в домашний каталог, уточнение имени пользователя

4. Уточняем имя пользователя, его группу, а также группы, куда входит пользователь, командой `id`. Сравниваем вывод `id` с выводом команды `groups` (рис. 4.4).

```

guest
[guest@aaastrakhanceva ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@aaastrakhanceva ~]$ groups
guest
[guest@aaastrakhanceva ~]$ cat /etc/passwd

```

Рис. 4.4: Вывод имени пользователя, его группы и т. д.

5. Просматриваем файл `/etc/passwd` командой `cat /etc/passwd | grep guest`. Находим в нём свою учётную запись. Сравниваем выводы команды `id`, `groups` и содержимое данного файла. Видим, что `uid` и `gid` пользователя везде совпадают (рис. 4.5).

```

guest:x:1001:1001::/home/guest:/bin/bash
[guest@aaastrakhanceva ~]$ cat /etc/passwd | grep guest
guest:x:1001:1001::/home/guest:/bin/bash
[guest@aaastrakhanceva ~]$

```

Рис. 4.5: Просмотр файла `/etc/passwd`

6. Определяем существующие в системе директории командой `ls -l /home/`. Нам удастся получить список поддиректорий директории `/home/`. В нём содержатся директории `'guest'` и `'user'`. Для обеих директорий установлены такие права: `'drwx---`'. Это говорит нам о том, что перед нами директории (первая буква `'d'`), и для этих директорий владелец (owner) единственный имеет права на чтение, запись и выполнение данных директорий (рис. 4.6).

```

[guest@aaastrakhanceva ~]$ ls -l /home/
total 8
drwx-----. 14 guest guest 4096 Mar  1 12:10 guest
drwx-----. 17 user  user  4096 Mar  1 12:09 user
[guest@aaastrakhanceva ~]$

```

Рис. 4.6: Просмотр поддиректорий `home`

7. Проверяем, какие расширенные атрибуты установлены на поддиректориях, находящихся в директории `/home`, командой: `lsattr /home`. Нам не удастся ни увидеть расширенные атрибуты директории, ни расширенные атрибуты директорий других пользователей (рис. 4.7).

```

drwxr-xr-x. 17 user user 4096 Mar 1 12:09 user
[guest@aaastrakhanceva ~]$ lsattr /home
lsattr: Permission denied While reading flags on /home/user
----- /home/guest
[guest@aaastrakhanceva ~]$

```

Рис. 4.7: Просмотр атрибутов

8. Создаем в домашней директории поддиректорию dir1 командой `mkdir dir1`. С помощью команд `ls -l` и `lsattr` определяем, какие права доступа и расширенные атрибуты были выставлены на директорию dir1. Можем видеть, что для данной директории выставленные стандартные для данного каталога права 4.8).

```

[guest@aaastrakhanceva ~]$ mkdir dir1
[guest@aaastrakhanceva ~]$ ls -l
total 0
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Desktop
drwxr-xr-x. 2 guest guest 6 Mar 1 12:22 dir1
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Documents
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Downloads
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Music
drwxr-xr-x. 2 guest guest 53 Mar 1 12:22 Pictures
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Public
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Templates
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Videos
[guest@aaastrakhanceva ~]$ lsattr
----- ./Desktop
----- ./Downloads
----- ./Templates
----- ./Public
----- ./Documents
----- ./Music
----- ./Pictures
----- ./Videos
----- ./dir1
[guest@aaastrakhanceva ~]$

```

Рис. 4.8: Создание новой директории и просмотр прав доступа на нее

9. Снимаем с директории dir1 все атрибуты командой `chmod 000 dir1` и проверяем результат с помощью выполнения команды `ls -l` (рис.4.9).

```
[guest@aaastrakhanceva ~]$ chmood 000 dir1
bash: chmood: command not found...
Similar command is: 'chmod'
[guest@aaastrakhanceva ~]$ chmod 000 dir1
[guest@aaastrakhanceva ~]$ ls -l
total 0
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Desktop
d------. 2 guest guest 6 Mar 1 12:22 dir1
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Documents
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Downloads
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Music
drwxr-xr-x. 2 guest guest 53 Mar 1 12:22 Pictures
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Public
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Templates
drwxr-xr-x. 2 guest guest 6 Mar 1 12:10 Videos
[guest@aaastrakhanceva ~]$
```

Рис. 4.9: Снятие всех атрибутов с созданной директории

10. Попытаемся создать в директории dir1 файл file1 командой `echo "test" > /home/guest/dir1/file1`. Получаем отказ в выполнении данной команды, так как на предыдущем шаге мы сняли все атрибуты с данной директории, таким образом теперь никакой пользователь не может ни прочитать файл, ни изменить его, ни запустить на выполнение до тех пор, пока права доступа не будут снова изменены (рис.4.10).

```
[guest@aaastrakhanceva ~]$ echo "test" > /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Permission denied
[guest@aaastrakhanceva ~]$ ls -l /home/guest/dir1
ls: cannot open directory '/home/guest/dir1': Permission denied
[guest@aaastrakhanceva ~]$
```

Рис. 4.10: Попытка создать файл в директории со снятыми атрибутами парв доступа

## 4.1 Заполнение таблицы 2.1

Права ди- ректо- рии	Права файла	Со- зда- ние файла	Уда- ление файла	За- пись в файл	Чте- ние файла	Сме- на ди- ректо- рии	Про- смотр фай- лов в ди- ректо- рии	Переиме- нование файла	Сме- на атри- бутов файла
d(000)	(000)	-	-	-	-	-	-	-	-
d(000)	(100)	-	-	-	-	-	-	-	-
d(000)	(200)	-	-	-	-	-	-	-	-
d(000)	(300)	-	-	-	-	-	-	-	-
d(000)	(400)	-	-	-	-	-	-	-	-
d(000)	(500)	-	-	-	-	-	-	-	-
d(000)	(600)	-	-	-	-	-	-	-	-
d(000)	(700)	-	-	-	-	-	-	-	-
d(100)	(000)	-	-	-	-	+	-	-	+
d(100)	(100)	-	-	-	-	+	-	-	+
d(100)	(200)	-	-	+	-	+	-	-	+
d(100)	(300)	-	-	+	-	+	-	-	+
d(100)	(400)	-	-	-	+	+	-	-	+
d(100)	(500)	-	-	-	+	+	-	-	+
d(100)	(600)	-	-	+	+	+	-	-	+
d(100)	(700)	-	-	+	+	+	-	-	+
d(200)	(000)	-	-	-	-	-	-	-	-
d(200)	(100)	-	-	-	-	-	-	-	-
d(200)	(200)	-	-	-	-	-	-	-	-
d(200)	(300)	-	-	-	-	-	-	-	-
d(200)	(400)	-	-	-	-	-	-	-	-

---

d(200)	(500)	-	-	-	-	-	-	-	-
d(200)	(600)	-	-	-	-	-	-	-	-
d(200)	(700)	-	-	-	-	-	-	-	-
d(300)	(000)	+	+	-	-	+	-	+	+
d(300)	(100)	+	+	-	-	+	-	+	+
d(300)	(200)	+	+	+	-	+	-	+	+
d(300)	(300)	+	+	+	-	+	-	+	+
d(300)	(400)	+	+	-	+	+	-	+	+
d(300)	(500)	+	+	-	+	+	-	+	+
d(300)	(600)	+	+	+	+	+	-	+	+
d(300)	(700)	+	+	+	+	+	-	+	+
d(400)	(000)	-	-	-	-	-	+	-	-
d(400)	(100)	-	-	-	-	-	+	-	-
d(400)	(200)	-	-	-	-	-	+	-	-
d(400)	(300)	-	-	-	-	-	+	-	-
d(400)	(400)	-	-	-	-	-	+	-	-
d(400)	(500)	-	-	-	-	-	+	-	-
d(400)	(600)	-	-	-	-	-	+	-	-
d(400)	(700)	-	-	-	-	-	+	-	-
d(500)	(000)	-	-	-	-	+	+	-	+
d(500)	(100)	-	-	-	-	+	+	-	+
d(500)	(200)	-	-	+	-	+	+	-	+
d(500)	(300)	-	-	+	-	+	+	-	+
d(500)	(400)	-	-	-	+	+	+	-	+
d(500)	(500)	-	-	-	+	+	+	-	+
d(500)	(600)	-	-	+	+	+	+	-	+
d(500)	(700)	-	-	+	+	+	+	-	+
d(600)	(000)	-	-	-	-	-	+	-	-

d(600)	(100)	-	-	-	-	-	+	-	-
d(600)	(200)	-	-	-	-	-	+	-	-
d(600)	(300)	-	-	-	-	-	+	-	-
d(600)	(400)	-	-	-	-	-	+	-	-
d(600)	(500)	-	-	-	-	-	+	-	-
d(600)	(600)	-	-	-	-	-	+	-	-
d(600)	(700)	-	-	-	-	-	+	-	-
d(700)	(000)	+	+	-	-	+	+	+	+
d(700)	(100)	+	+	-	-	+	+	+	+
d(700)	(200)	+	+	+	-	+	+	+	+
d(700)	(300)	+	+	+	-	+	+	+	+
d(700)	(400)	+	+	-	+	+	+	+	+
d(700)	(500)	+	+	-	+	+	+	+	+
d(700)	(600)	+	+	+	+	+	+	+	+
d(700)	(700)	+	+	+	+	+	+	+	+

Таблица 2.1 «Установленные права и разрешённые действия»

## 4.2 Заполнение таблицы 2.2

Операция	Минималь- ные права на директорию	Минималь- ные права на файл
Создание файла	d(300)	-
Удаление файла	d(300)	-
Чтение файла	d(100)	(400)

Запись в файл	d(100)	(200)
Переименование файла	d(300)	(000)
Создание поддиректории	d(300)	-
Удаление поддиректории	d(300)	-

Таблица 2.2 “Минимальные права для совершения операций”



## 5 Выводы

В ходе выполнения данной лабораторной работы я получила практические навыки работы в консоли с атрибутами файлов, закрепила теоретические основы дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux.

## 6 Список литературы. Библиография

[1] Права доступа в Linux: <https://codechick.io/tutorials/unix-linux/unix-linux-permissions> [2] Курс “Основы информационной безопасности”: <https://esystem.rudn.ru/course/view.php?id=6078>