

# **Отчёт по лабораторной работе №4**

**дисциплина: Информационная безопасность**

**Быстров Глеб Андреевич**

# 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>10</b>
<b>5</b>	<b>Выводы</b>	<b>14</b>
	<b>Список литературы</b>	<b>15</b>

## Список иллюстраций

4.1	Расширенные атрибуты файла . . . . .	10
4.2	Установка прав . . . . .	10
4.3	Установка расширенного атрибута . . . . .	10
4.4	Установка расширенного атрибута . . . . .	11
4.5	Проверка . . . . .	11
4.6	Запись в файл . . . . .	11
4.7	Попытка удаления . . . . .	12
4.8	Попытка установки прав . . . . .	12
4.9	Снятие атрибута и новая попытка . . . . .	12
4.10	Повтор действий с атрибутом «i» . . . . .	13

## Список таблиц

# 1 Цель работы

В данной лабораторной работе мне будет необходимо получить практические навыки работы в консоли с расширенными атрибутами файлов.

## 2 Задание

Последовательно выполнить пункты по работе с расширенными атрибутами в терминале.

### 3 Теоретическое введение

В Linux у каждого файла и каждого каталога есть два владельца: пользователь и группа.

Эти владельцы устанавливаются при создании файла или каталога. Пользователь, который создаёт файл становится владельцем этого файла, а первичная группа, в которую входит этот же пользователь, так же становится владельцем этого файла. Чтобы определить, есть ли у вас как у пользователя права доступа к файлу или каталогу, оболочка проверяет владение ими. [1].

Выделяют три категории пользователей, которым могут предоставляться права на файл:

- Сам владелец (u – user) объекта – конкретный пользователь, чье имя числится в атрибутах файла как имя владельца этого файла. Обычно если пользователь создает файл, то он автоматически записывается как его владелец.
- Группа (g – group), к которой принадлежит владелец файла. Когда в Linux создается пользователь, то для него создается одноименная группа. Однако средствами администрирования системы можно объединять пользователей в различные группы. При этом конкретный пользователь может входить в состав нескольких групп. Группы позволяют предоставлять права доступа к ресурсам сразу нескольким людям, но при этом ограниченному кругу лиц.
- Все остальные (o – other) – это все те, кто не является владельцем файла и не принадлежит к группе владельца файла. То есть любой посторонний пользователь.

Чтение, запись, выполнение – это то, что можно делать с существующим файлом, возможные действия над ним. У каждой категории пользователей (владельца, группы, остальных) должны быть свои права на каждое вышеупомянутое действие.

- Право на чтение (r – read) означает, что файл можно просматривать. Например, открыть файл и, если он текстовый, прочитать содержащийся в нем текст. Если это файл изображения, то можно посмотреть изображение. Наличие права только на чтение не позволяет изменять файл. То есть нельзя будет исправить текст или подрисовать что-то к картинке.
- Право на запись (w – write) позволяет изменять файл, то есть дописывать в него информацию или заменять ее другой.
- Право на исполнение (x – execution) имеет смысл не для всех файлов, хотя может быть установлено для любого. Это право позволяет исполнять файл как программу, при этом в файле должны быть записаны инструкции для процессора, то есть файл должен быть исполняемой программой.

Первые три записи – это права владельца, вторые три записи – права группы, последняя тройка – права на файл для всех остальных. Если обозначить каждое право соответствующей буквой, и все права всем предоставляются, то получится такая запись: `rw xrwxrwx` [2].

Рассмотрим подробнее, что значат условные значения флагов прав:

- - нет прав, совсем;
- x - разрешено только выполнение файла, как программы но не изменение и не чтение;
- w- - разрешена только запись и изменение файла;
- wx - разрешено изменение и выполнение, но в случае с каталогом, вы не можете посмотреть его содержимое;
- r- - права только на чтение;



r-x - только чтение и выполнение, без права на запись;  
rw- - права на чтение и запись, но без выполнения;  
rwx - все права;  
-s - установлен SUID или SGID бит, первый отображается в поле для владельца, второй для группы;  
-t - установлен sticky-bit, а значит пользователи не могут удалить этот файл [3].

Использование команды ls с опцией -l выведет на экран «длинную» распечатку, в которой будут, среди прочего, отражены права доступа к файлу [4].

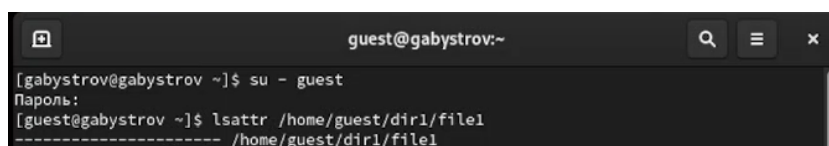
Все группы, созданные в системе, находятся в файле /etc/group. Посмотрев содержимое этого файла, вы можете узнать список групп linux, которые уже есть в вашей системе.

Кроме стандартных root и users, здесь есть еще пару десятков групп. Это группы, созданные программами, для управления доступом этих программ к общим ресурсам. Каждая группа разрешает чтение или запись определенного файла или каталога системы, тем самым регулируя полномочия пользователя, а следовательно, и процесса, запущенного от этого пользователя. Здесь можно считать, что пользователь - это одно и то же что процесс, потому что у процесса все полномочия пользователя, от которого он запущен [5].

Расширенные атрибуты файловых объектов (далее - расширенные атрибуты) - поддерживаемая некоторыми файловыми системами возможность ассоциировать с файловыми объектами произвольные метаданные. В отличие от обычных атрибутов файловых объектов (таких, как владелец, права доступа, время создания и пр.), содержание расширенных атрибутов не специфицируется в файловой системе и может принимать любые значения. С точки зрения реализации расширенные атрибуты представляют собой пары ключ:значение, ассоциированные с файловыми объектами. Типичными применениями расширенных атрибутов является хранение таких данных, как автор документа, контрольные суммы, источник документа, информация для контроля доступа [6].

## 4 Выполнение лабораторной работы

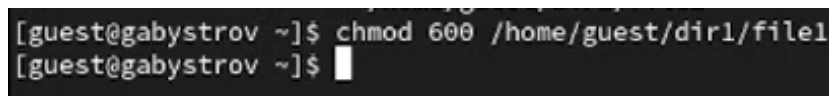
1. От имени пользователя guest определите расширенные атрибуты файла /home/guest/dir1/file1 командой lsattr /home/guest/dir1/file1 (рис. 4.1).



```
guest@gabystrov:~  
[gabystrov@gabystrov ~]$ su - guest  
Пароль:  
[guest@gabystrov ~]$ lsattr /home/guest/dir1/file1  
----- /home/guest/dir1/file1
```

Рис. 4.1: Расширенные атрибуты файла

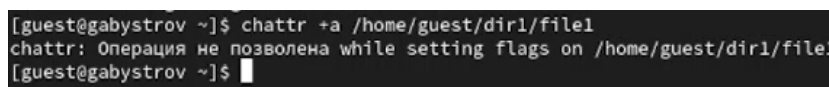
2. Установил командой chmod 600 file1 на файл file1 права, разрешающие чтение и запись для владельца файла (рис. 4.2).



```
[guest@gabystrov ~]$ chmod 600 /home/guest/dir1/file1  
[guest@gabystrov ~]$
```

Рис. 4.2: Установка прав

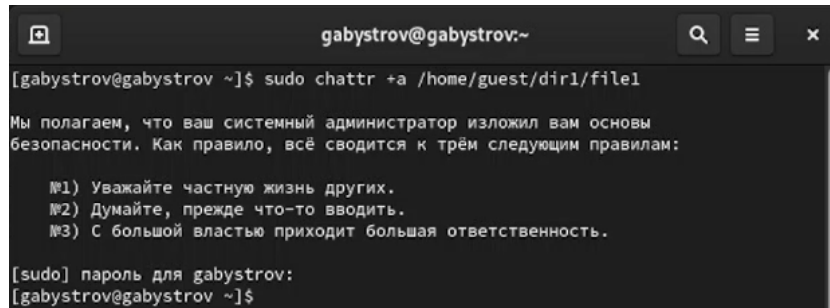
3. Попробовал установить на файл /home/guest/dir1/file1 расширенный атрибут а от имени пользователя guest (рис. 4.3).



```
[guest@gabystrov ~]$ chattr +a /home/guest/dir1/file1  
chattr: Операция не позволена while setting flags on /home/guest/dir1/file1  
[guest@gabystrov ~]$
```

Рис. 4.3: Установка расширенного атрибута

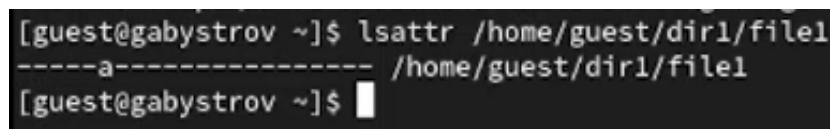
4. Повысил свои права с помощью команды `su`. Попробовал установить расширенный атрибут `a` на файл `/home/guest/dir1/file1` от имени суперпользователя (рис. 4.4).



```
gabystrov@gabystrov:~  
[gabystrov@gabystrov ~]$ sudo chattr +a /home/guest/dir1/file1  
Мы полагаем, что ваш системный администратор изложил вам основы  
безопасности. Как правило, всё сводится к трём следующим правилам:  
  
№1) Уважайте частную жизнь других.  
№2) Думайте, прежде что-то вводить.  
№3) С большой властью приходит большая ответственность.  
  
[sudo] пароль для gabystrov:  
[gabystrov@gabystrov ~]$
```

Рис. 4.4: Установка расширенного атрибута

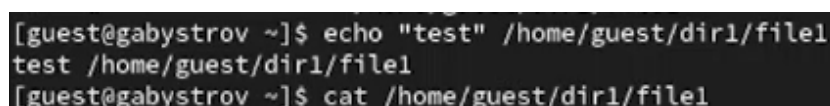
5. От пользователя `guest` проверил правильность установления атрибута (рис. 4.5).



```
[guest@gabystrov ~]$ lsattr /home/guest/dir1/file1  
-----a----- /home/guest/dir1/file1  
[guest@gabystrov ~]$
```

Рис. 4.5: Проверка

6. Выполнил дозапись в файл `file1` слова «test» (рис. 4.6).



```
[guest@gabystrov ~]$ echo "test" /home/guest/dir1/file1  
test /home/guest/dir1/file1  
[guest@gabystrov ~]$ cat /home/guest/dir1/file1  
test
```

Рис. 4.6: Запись в файл

7. Попробовал удалить файл `file1` либо стереть имеющуюся в нём информацию (рис. 4.7).

```
[guest@gabystrov ~]$ echo "abcd" > /home/guest/dirl/file1
-bash: /home/guest/dirl/file1: Операция не позволена
[guest@gabystrov ~]$ rename file1 file2 /home/guest/dirl/file1
rename: /home/guest/dirl/file1: не удалось переименовать в /home/guest/dirl/file2: Операция не позволена
```

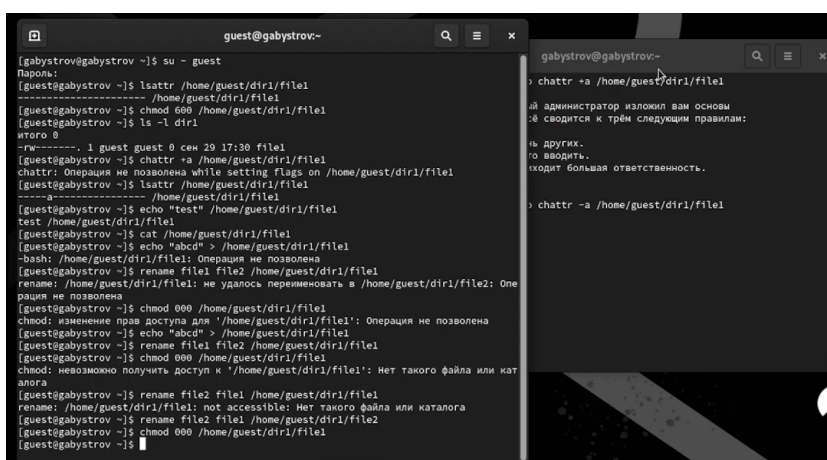
Рис. 4.7: Попытка удаления

8. Попробовал с помощью команды установить на файл file1 права, например, запрещающие чтение и запись для владельца файла (рис. 4.8).

```
[guest@gabystrov ~]$ chmod 000 /home/guest/dirl/file1
chmod: изменение прав доступа для '/home/guest/dirl/file1': Операция не позволена
[guest@gabystrov ~]$
```

Рис. 4.8: Попытка установки прав

9. Снял расширенный атрибут a с файла /home/guest/dirl/file1 от имени супер-пользователя. Повторил операции, которые ранее не удавалось выполнить (рис. 4.9).



```
[gabystrov@gabystrov ~]$ su - guest
[guest@gabystrov ~]$ lsattr /home/guest/dirl/file1
----- /home/guest/dirl/file1
[guest@gabystrov ~]$ chmod 600 /home/guest/dirl/file1
[guest@gabystrov ~]$ ls -l dirl
total 0
-rw-r--r-- 1 guest guest 0 сен 29 17:30 file1
[guest@gabystrov ~]$ chattr +a /home/guest/dirl/file1
chattr: Операция не позволена while setting flags on /home/guest/dirl/file1
[guest@gabystrov ~]$ lsattr /home/guest/dirl/file1
----- /home/guest/dirl/file1
[guest@gabystrov ~]$ echo "test" /home/guest/dirl/file1
test /home/guest/dirl/file1
[guest@gabystrov ~]$ cat /home/guest/dirl/file1
[guest@gabystrov ~]$ echo "abcd" > /home/guest/dirl/file1
-bash: /home/guest/dirl/file1: Операция не позволена
[guest@gabystrov ~]$ rename file1 file2 /home/guest/dirl/file1
rename: /home/guest/dirl/file1: не удалось переименовать в /home/guest/dirl/file2: Операция не позволена
[guest@gabystrov ~]$ chmod 000 /home/guest/dirl/file1
chmod: изменение прав доступа для '/home/guest/dirl/file1': Операция не позволена
[guest@gabystrov ~]$ echo "abcd" > /home/guest/dirl/file1
[guest@gabystrov ~]$ rename file1 file2 /home/guest/dirl/file1
rename: /home/guest/dirl/file1: не удалось переименовать в /home/guest/dirl/file2: Операция не позволена
[guest@gabystrov ~]$ chown 000 /home/guest/dirl/file1
chown: невозможно получить доступ к '/home/guest/dirl/file1': Нет такого файла или каталога
[guest@gabystrov ~]$ rename file2 file1 /home/guest/dirl/file1
rename: /home/guest/dirl/file1: not accessible: Нет такого файла или каталога
[guest@gabystrov ~]$ rename file2 file1 /home/guest/dirl/file2
[guest@gabystrov ~]$ chown 000 /home/guest/dirl/file1
[guest@gabystrov ~]$ su -
[root@gabystrov ~]# chattr -a /home/guest/dirl/file1
[root@gabystrov ~]# chattr -a /home/guest/dirl/file1
```

Рис. 4.9: Снятие атрибута и новая попытка

10. Повторил действия по шагам, заменив атрибут «a» атрибутом «i». Проверил можно ли дозаписать информацию в файл (рис. 4.10).

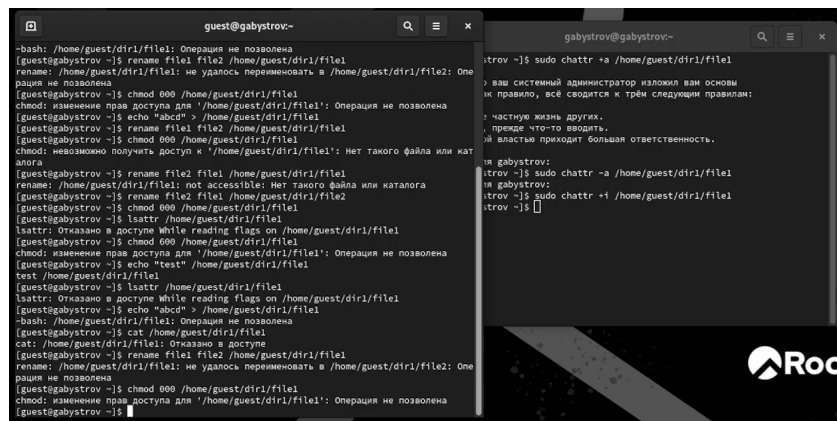


Рис. 4.10: Повтор действий с атрибутом «i»

## 5 Выводы

В данной лабораторной работе мне успешно удалось получить практические навыки работы в консоли с расширенными атрибутами файлов.

## Список литературы

1. Права в Linux (chown, chmod, SUID, GUID, sticky bit, ACL, umask) [Электронный ресурс]. 2023. URL: <https://habr.com/ru/articles/469667/>.
2. Права доступа к файлам и каталогам [Электронный ресурс]. 2023. URL: <https://younglinux.info/bash/rwx>.
3. Права доступа к файлам в Linux [Электронный ресурс]. 2023. URL: <https://losst.pro/prava-dostupa-k-fajlam-v-linux>.
4. Права доступа к файлам [Электронный ресурс]. 2023. URL: <https://docs.altlinux.org/ru-RU/archive/2.3/html-single/junior/alt-docs-extras-linuxnovice/ch02s08.html>.
5. Группы пользователей Linux [Электронный ресурс]. 2023. URL: <https://losst.pro/gruppy-polzovatelej-linux>.
6. Работа с расширенными атрибутами [Электронный ресурс]. 2023. URL: <https://wiki.astralinux.ru/pages/viewpage.action?pageId=149063848>.