

**РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ**

**Факультет физико-математических и естественных наук**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**

*дисциплина: Основы информационной безопасности*

Студент: Накова Амина Михайловна

Студ. билет № 1132232887

Группа: НПИбд-02-23

**МОСКВА**

2025 г.

## Цель работы:

Получение практических навыков работы в консоли с расширенными атрибутами файлов

## Выполнение работы:

1. От имени пользователя guest определите расширенные атрибуты файла /home/guest/dir1/file1 командой

```
[guest@localhost ~]$ lsattr /home/guest/dir1/file1
----i----- /home/guest/dir1/file1
```

Рис. 1.1.

2. Установлю командой chmod 600 file1 на файл file1 права, разрешающие чтение и запись для владельца файла.

```
[guest@localhost ~]$ chmod 600 file1
```

Рис. 2.1.

3. Попробую установить на файл /home/guest/dir1/file1 расширенный атрибут a от имени пользователя guest: chattr +a /home/guest/dir1/file1. В ответ вы должны получить отказ от выполнения операции

```
[guest@localhost ~]$ chattr +a /home/guest/dir1/file1
chattr: Отказано в доступе while trying to stat /home/guest/dir1/file1
```

Рис. 3.1.

4. Зайдите на третью консоль с правами администратора либо повышу свои права с помощью команды su. Попробуйте установить расширенный атрибут a на файл /home/guest/dir1/file1 от имени суперпользователя: chattr +a /home/guest/dir1/file1

```
[guest@localhost ~]$ su - amina
Пароль:
[amina@localhost ~]$ sudo -i
[sudo] пароль для amina:
[root@localhost ~]# chatter +a /home/guest/dir1/file1
```

Рис. 4.1.

5. Попытаюсь создать в директории dir1 файл file1 командой echo "test" > /home/guest/dir1/file1. Проверю командой ls -l /home/guest/dir1

```
[guest@localhost ~]$ echo "test"
test
[guest@localhost ~]$ /home/guest/dir1/file1
-bash: /home/guest/dir1/file1: Отказано в доступе
[guest@localhost ~]$ ls -l /home/guest/dir1
ls: невозможно открыть каталог '/home/guest/dir1': Отказано в доступе
```

Рис. 5.1.

6. От пользователя guest проверьте правильность установления атрибута: lsattr /home/guest/dir1/file1. Выполните дозапись в файл file1 слова «test» командой echo "test" /home/guest/dir1/file1. После этого выполните чтение файла file1 командой. cat /home/guest/dir1/file1. Убедитесь, что слово test было успешно записано в file1. Попробуйте удалить файл file1 либо стереть имеющуюся в нём информацию командой echo "abcd" > /home/guest/dir1/file1. Попробуйте переименовать файл. Попробуйте с помощью команды chmod 000 file1 установить на файл file1 права, например, запрещающие чтение и запись для владельца файла. Удалось ли вам успешно выполнить указанные команды? Снимите расширенный атрибут a с файла /home/guest/dir1/file1 от имени суперпользователя командой chatter -a /home/guest/dir1/file1. Повторите операции, которые вам ранее не удавалось выполнить.

```
[root@localhost ~]# chattr -a /home/guest/dir1/file1
[root@localhost ~]# su - guest
[guest@localhost ~]$ cat /home/guest/dir1/file1
cat: /home/guest/dir1/file1: Отказано в доступе
[guest@localhost ~]$ su - amina
Пароль:
[amina@localhost ~]$ sudo cat /home/guest/dir1/file1
abcd
```

Рис. 6. 1

### **Вывод:**

В ходе лабораторной работы были получены практические навыки работы с расширенными атрибутами файлов в Linux. На примерах было изучено действие двух ключевых атрибутов:

1. Атрибут `a` (`append only`): Разрешает только операцию дозаписи в конец файла. Любые другие операции, изменяющие содержимое файла (перезапись) или его метаданные (удаление, переименование), блокируются. Изменение базовых прав доступа (`chmod`) остается возможным.
2. Атрибут `i` (`immutable`): Делает файл полностью неизменяемым. Запрещаются все операции по изменению файла: запись (как дозапись, так и перезапись), удаление, переименование и изменение прав доступа. Это максимальный уровень защиты содержимого файла от изменений.

Было установлено, что для управления расширенными атрибутами необходимы права суперпользователя. Расширенные атрибуты являются мощным механизмом разграничения доступа, который действует на уровне файловой системы и применяется поверх классической дискреционной модели прав (права `gwx`). Это позволяет администратору установить более строгие ограничения, которые не могут быть обойдены простым изменением базовых прав доступа.