

Лабораторная работа №4

Основы информационной безопасности

Сабралиева Марворид Нуралиевна

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	10
	Список литературы	11

Список иллюстраций

2.1	Установка прав	6
2.2	Расширенный атрибут а	7
2.3	Действия с файлом	8
2.4	Расширенный атрибут а	8
2.5	Расширенный атрибут і	9

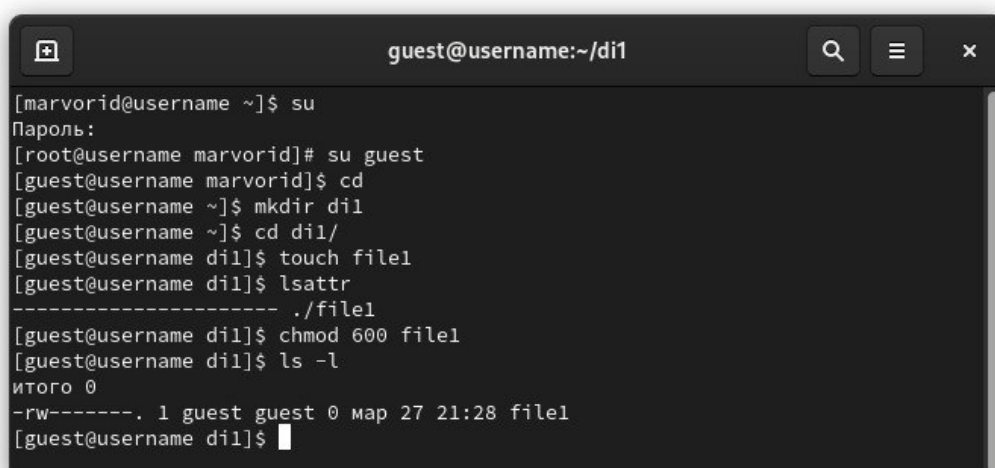
Список таблиц

1 Цель работы

Получение практических навыков работы в консоли с расширенными атрибутами файлов.

2 Выполнение лабораторной работы

1. От имени пользователя guest определим расширенные атрибуты файла /home/guest/dir1/file1 командой `lsattr /home/guest/dir1/file1`
2. Установим командой `chmod 600 file1` на файл file1 права, разрешающие чтение и запись для владельца файла. (рис. 2.1).



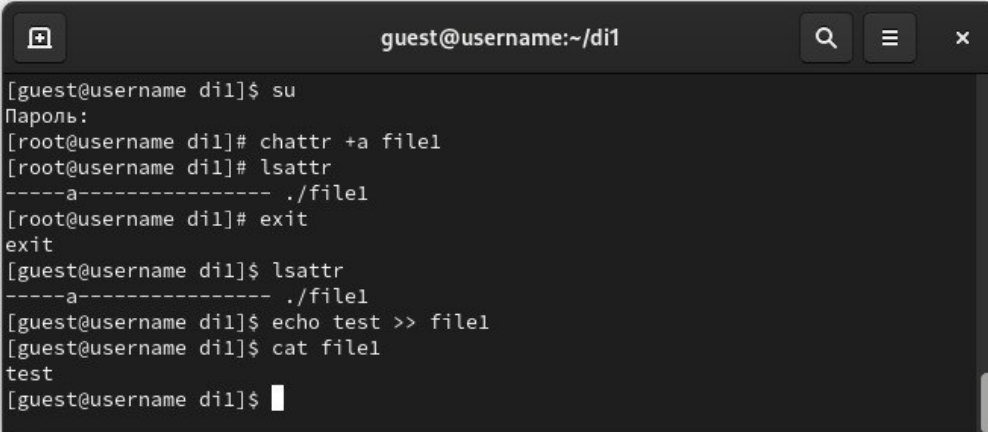
```
guest@username:~/di1
[marvorid@username ~]$ su
Пароль:
[root@username marvorid]# su guest
[guest@username marvorid]$ cd
[guest@username ~]$ mkdir di1
[guest@username ~]$ cd di1/
[guest@username di1]$ touch file1
[guest@username di1]$ lsattr
----- ./file1
[guest@username di1]$ chmod 600 file1
[guest@username di1]$ ls -l
итого 0
-rw-----. 1 guest guest 0 map 27 21:28 file1
[guest@username di1]$
```

Рис. 2.1: Установка прав

3. Попробуем установить на файл /home/guest/dir1/file1 расширенный атрибут а от имени пользователя guest: `chattr +a /home/guest/dir1/file1` В ответ мы получили отказ от выполнения операции.
4. Зайдем на третью консоль с правами администратора либо можно повысить свои права с помощью команды `su`. Попробуем установить расширенный

атрибут `a` на файл `/home/guest/dir1/file1` от имени суперпользователя: `chattr +a /home/guest/dir1/file1` Команда выполнена

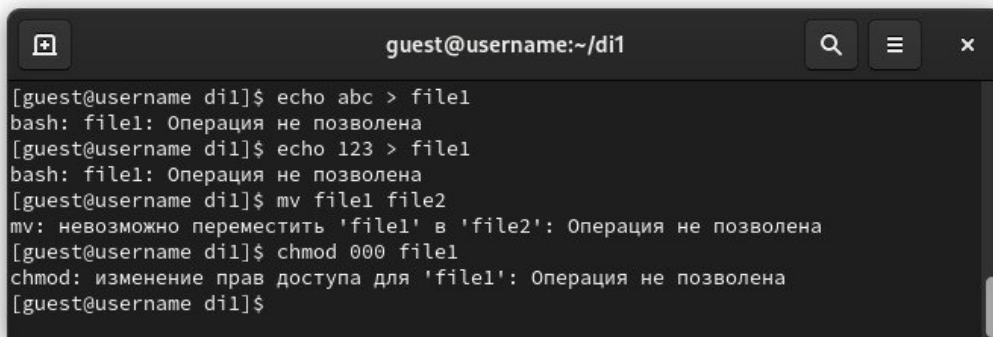
5. От пользователя `guest` проверим правильность установления атрибута: `lsattr`
6. Выполните дозапись в файл `file1` слова «test» командой `echo "test" >> file1` После этого выполним чтение файла `file1` командой `cat file1` Проверим, что слово `test` было успешно записано в `file1`. (рис. 2.2).



```
guest@username:~/di1
[guest@username di1]$ su
Пароль:
[root@username di1]# chattr +a file1
[root@username di1]# lsattr
-----a----- ./file1
[root@username di1]# exit
exit
[guest@username di1]$ lsattr
-----a----- ./file1
[guest@username di1]$ echo test >> file1
[guest@username di1]$ cat file1
test
[guest@username di1]$
```

Рис. 2.2: Расширенный атрибут `a`

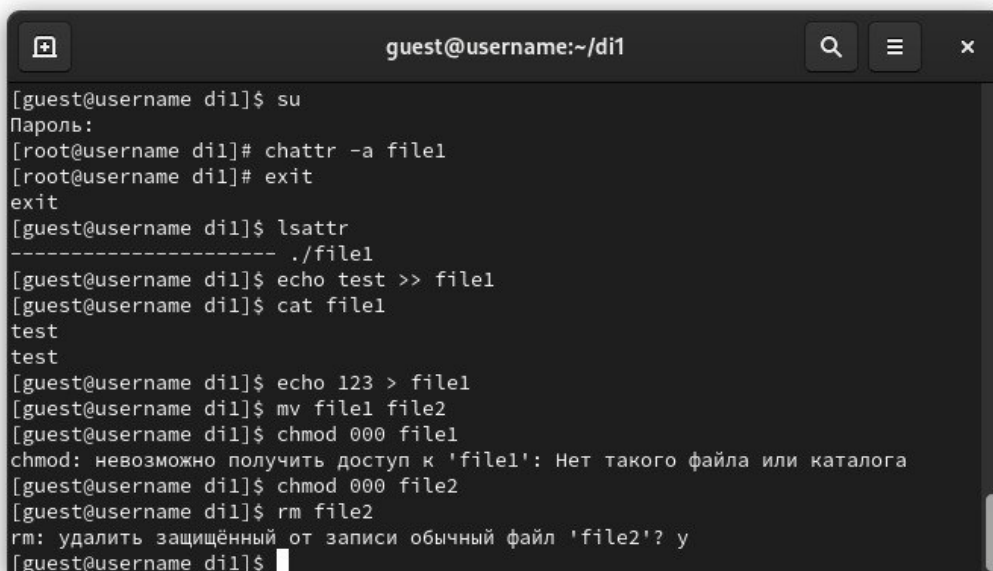
7. Попробуем удалить файл `file1` либо стереть имеющуюся в нём информацию командой `echo "abcd" > file1` Попробуем переименовать файл. Ничего из этого не выполняется
8. Попробуем с помощью команды `chmod 000 file1` установить на файл `file1` права, например, запрещающие чтение и запись для владельца файла. Данная команда тоже не сработала (рис. 2.3).



```
guest@username:~/di1
[guest@username di1]$ echo abc > file1
bash: file1: Операция не позволена
[guest@username di1]$ echo 123 > file1
bash: file1: Операция не позволена
[guest@username di1]$ mv file1 file2
mv: невозможно переместить 'file1' в 'file2': Операция не позволена
[guest@username di1]$ chmod 000 file1
chmod: изменение прав доступа для 'file1': Операция не позволена
[guest@username di1]$
```

Рис. 2.3: Действия с файлом

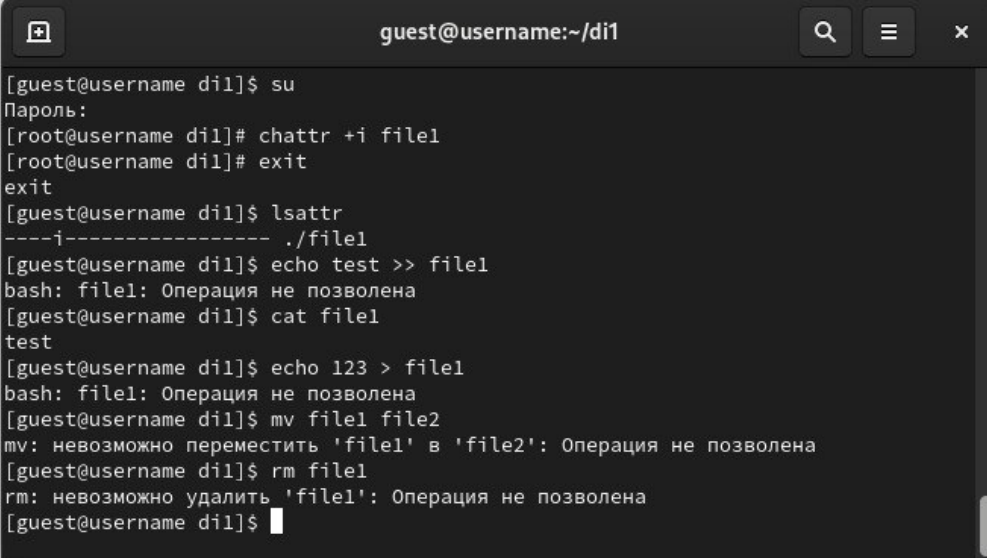
9. Снимем расширенный атрибут `a` с файла `/home/guest/dirl/file1` от имени суперпользователя командой `chattr -a file1`. Повторим операции, которые нам ранее не удавалось выполнить. После снятия атрибута `-a` стало возможно переписать файл, удалить или переименовать. Атрибут `-a` позволяет только дозаписывать файл (рис. 2.4).



```
guest@username:~/di1
[guest@username di1]$ su
Пароль:
[root@username di1]# chattr -a file1
[root@username di1]# exit
exit
[guest@username di1]$ lsattr
----- ./file1
[guest@username di1]$ echo test >> file1
[guest@username di1]$ cat file1
test
test
[guest@username di1]$ echo 123 > file1
[guest@username di1]$ mv file1 file2
[guest@username di1]$ chmod 000 file1
chmod: невозможно получить доступ к 'file1': Нет такого файла или каталога
[guest@username di1]$ chmod 000 file2
[guest@username di1]$ rm file2
rm: удалить защищённый от записи обычный файл 'file2'? y
[guest@username di1]$
```

Рис. 2.4: Расширенный атрибут `a`

10. Повторим наши действия по шагам, заменив атрибут «a» атрибутом «i». Удалось ли вам дозаписать информацию в файл? Атрибут -i запрещает любое изменение файла: дозапись, переименование, удаление, смену атрибутов (рис. 2.5).



```
guest@username:~/di1
[guest@username di1]$ su
Пароль:
[root@username di1]# chattr +i file1
[root@username di1]# exit
exit
[guest@username di1]$ lsattr
----i----- ./file1
[guest@username di1]$ echo test >> file1
bash: file1: Операция не позволена
[guest@username di1]$ cat file1
test
[guest@username di1]$ echo 123 > file1
bash: file1: Операция не позволена
[guest@username di1]$ mv file1 file2
mv: невозможно переместить 'file1' в 'file2': Операция не позволена
[guest@username di1]$ rm file1
rm: невозможно удалить 'file1': Операция не позволена
[guest@username di1]$
```

Рис. 2.5: Расширенный атрибут i

3 Выводы

В результате выполнения работы мы повысили свои навыки использования интерфейса командой строки (CLI), познакомились на примерах с тем, как используются основные и расширенные атрибуты при разграничении доступа. Имели возможность связать теорию дискреционного разделения доступа (дискреционная политика безопасности) с её реализацией на практике в ОС Linux. Опробовали действие на практике расширенных атрибутов «а» и «і»

Список литературы