

**Отчёт по лабораторной работе №4.
Дискреционное разграничение прав в
Linux. Расширенные атрибуты.**

Предмет: информационная безопасность

Александр Сергеевич Баклашов

Содержание

1	Цель работы	4
2	Теоретическое введение	5
3	Выполнение лабораторной работы	6
4	Вывод	10
5	Библиография	11

List of Figures

3.1	Атрибуты и права	6
3.2	Расш. атр.	6
3.3	su +a	7
3.4	Выполнение команд с +a	8
3.5	Снятем атрибута a	8
3.6	Команды без атр. a	9
3.7	+i	9
3.8	Выполнение команд с +i	9

1 Цель работы

Получение практических навыков работы в консоли с расширенными атрибутами файлов. [1]

2 Теоретическое введение

Информационная безопасность – это защищенность информации и поддерживающей инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, чреватых нанесением ущерба владельцам или пользователям информации и поддерживающей инфраструктуры.

`chmod` (от англ. `change mode`) — команда для изменения прав доступа к файлам и каталогам, используемая в Unix-подобных операционных системах. Входит в стандарт POSIX, в Coreutils. [3]

3 Выполнение лабораторной работы

1. От имени пользователя guest определим расширенные атрибуты файла /home/guest/dir1/file1 командой. Установим командой chmod 600 file1 на файл file1 права, разрешающие чтение и запись для владельца файла. (рис. 3.1)

```
[guest@asbaklashov ~]$ lsattr /home/guest/dir1/file1
----- /home/guest/dir1/file1
[guest@asbaklashov ~]$ chmod 600 dir1/file1
[guest@asbaklashov ~]$
```

Figure 3.1: Атрибуты и права

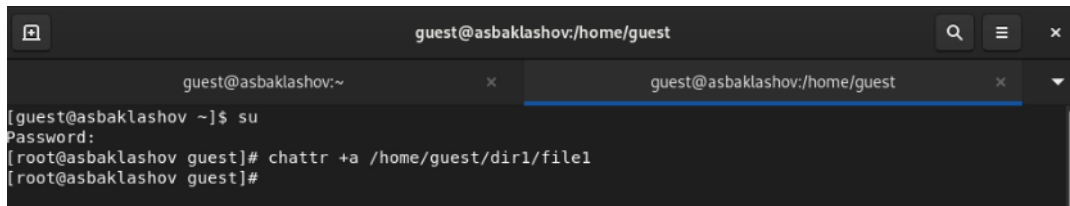
2. Попробуем установить на файл /home/guest/dir1/file1 расширенный атрибут а от имени пользователя guest: chattr +a /home/guest/dir1/file1 (рис. 3.2)

```
[guest@asbaklashov ~]$ chattr +a /home/guest/dir1/file1
chattr: Operation not permitted while setting flags on /home/guest/dir1/file1
[guest@asbaklashov ~]$
```

Figure 3.2: Расш. атр.

В ответ получили отказ от выполнения операции.

3. Зайдём на третью консоль с правами администратора либо повысим свои права с помощью команды su. Попробуем установить расширенный атрибут а на файл /home/guest/dir1/file1 от имени суперпользователя: chattr +a /home/guest/dir1/file1 (рис. 3.3)



```
guest@asbaklashov:/home/guest
[guest@asbaklashov ~]$ su
Password:
[root@asbaklashov guest]# chattr +a /home/guest/dir1/file1
[root@asbaklashov guest]#
```

Figure 3.3: su +a

4. От пользователя guest проверим правильность установления атрибута:
lsattr /home/guest/dir1/file1.

Выполним дозапись в файл file1 слова «test» командой echo “test” »
/home/guest/dir1/file1

После этого выполним чтение файла file1 командой cat /home/guest/dir1/file1

Убедимся, что слово test было успешно записано в file1.

Попробуем удалить файл file1 либо стереть имеющуюся в нём информацию командой echo “abcd” > /home/guest/dir1/file1

Попробуем переименовать файл.

Попробуем с помощью команды chmod 000 file1 установить на файл file1 права, например, запрещающие чтение и запись для владельца файла. (рис. 3.4)


```
[guest@asbaklashov ~]$ echo "abcd" > /home/guest/dir1/file1
[guest@asbaklashov ~]$ rename file1 file2 /home/guest/dir1/file1
[guest@asbaklashov ~]$ cat /home/guest/dir1/file2
abcd
[guest@asbaklashov ~]$ chmod 000 /home/guest/dir1/file2
[guest@asbaklashov ~]$ ls -l dir1
total 4
-----, 1 guest guest 5 Sep 25 13:43 file2
```

Figure 3.6: Команды без атр. а

7. Повторим действия по шагам, заменив атрибут «а» атрибутом «i». Удалось ли дозаписать информацию в файл? Наблюдения занесём в отчёт. (рис. 3.7, рис. 3.8)

```
[root@asbaklashov guest]# chatter +i /home/guest/dir1/file1
[root@asbaklashov guest]#
```

Figure 3.7: +i

```
[guest@asbaklashov ~]$ lsattr /home/guest/dir1/file1
----i----- /home/guest/dir1/file1
[guest@asbaklashov ~]$ echo "test" >> /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Operation not permitted
[guest@asbaklashov ~]$ echo "test" /home/guest/dir1/file1
test /home/guest/dir1/file1
[guest@asbaklashov ~]$ cat /home/guest/dir1/file1
abcd
[guest@asbaklashov ~]$ echo "test" >> /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Operation not permitted
[guest@asbaklashov ~]$ echo "abcde" > /home/guest/dir1/file1
bash: /home/guest/dir1/file1: Operation not permitted
[guest@asbaklashov ~]$ rename file1 file2 /home/guest/dir1/file1
rename: /home/guest/dir1/file1: rename to /home/guest/dir1/file2 failed: Operation not permitted
[guest@asbaklashov ~]$ chmod 000 /home/guest/dir1/file1
chmod: changing permissions of '/home/guest/dir1/file1': Operation not permitted
[guest@asbaklashov ~]$
```

Figure 3.8: Выполнение команд с +i

При применении расширенного атрибута “i” ни одного из действий, представленных в лабораторной работе, совершить не удалось.

4 Вывод

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

5 Библиография

1. Лабораторная работа №4. Дискреционное разграничение прав в Linux. Расширенные атрибуты. - 4 с. [Электронный ресурс]. М. URL: Лабораторная работа №4 (Дата обращения: 25.09.2022).
2. Rocky Linux Documentation. [Электронный ресурс]. М. URL: Rocky Linux Documentation (Дата обращения: 25.09.2022).
3. Chmod. [Электронный ресурс]. М. URL: Chmod (Дата обращения: 25.09.2022).