

Отчёт по лабораторной работе №4

Основы информационной безопасности

Мурашов Иван Вячеславович

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	9
5	Цель работы	10
6	Теоретическое введение	11
7	Выполнение лабораторной работы	13
8	Выводы	18

Список иллюстраций

7.1	Определение атрибутов	13
7.2	Изменение прав доступа	13
7.3	Попытка установки расширенных атрибутов	14
7.4	Установка расширенных атрибутов	14
7.5	Проверка атрибутов	14
7.6	Дозапись в файл	14
7.7	Попытка удалить файл	15
7.8	Попытка переименовать файл	15
7.9	Попытка изменить права доступа	15
7.10	Снятие расширенных атрибутов	15
7.11	Проверка выполнения действий	16
7.12	Попытка добавить расширенный атрибут	16
7.13	Добавление расширенного атрибута	16
7.14	Проверка выполнения действий	17

Список таблиц

1 Цель работы

Целью данной работы является приобретение практических навыков работы в консоли с атрибутами файлов для групп пользователей.

2 Задание

1. Создание пользователя guest2, добавление его в группу пользователей guest
2. Заполнение таблицы 3.1
3. Заполнение таблицы 3.2 на основе таблицы 3.1.

3 Теоретическое введение

Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенными файлами и каталогами. С помощью разрешений можно создать надежную среду — такую, в которой никто не может поменять содержимое ваших документов или повредить системные файлы. [1]

Группы пользователей Linux 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. [2]

- daemon - от имени этой группы и пользователя daemon запускаются сервисы, которым необходима возможность записи файлов на диск.
- sys - группа открывает доступ к исходникам ядра и файлам - include сохраненным в системе
- sync - позволяет выполнять команду /bin/sync
- games - разрешает играм записывать свои файлы настроек и историю в определенную папку
- man - позволяет добавлять страницы в директорию /var/cache/man
- lp - позволяет использовать устройства параллельных портов
- mail - позволяет записывать данные в почтовые ящики /var/mail/

- `proxy` - используется прокси серверами, нет доступа записи файлов на диск
- `www-data` - с этой группой запускается веб-сервер, она дает доступ на запись `/var/www`, где находятся файлы веб-документов
- `list` - позволяет просматривать сообщения в `/var/mail`
- `nogroup` - используется для процессов, которые не могут создавать файлов на жестком диске, а только читать, обычно применяется вместе с пользователем `nobody`.
- `adm` - позволяет читать логи из директории `/var/log`
- `tty` - все устройства `/dev/vcsa` разрешают доступ на чтение и запись пользователям из этой группы
- `disk` - открывает доступ к жестким дискам `/dev/sd*` `/dev/hd*`, можно сказать, что это аналог `root` доступа.
- `dialout` - полный доступ к серийному порту
- `cdrom` - доступ к CD-ROM
- `wheel` - позволяет запускать утилиту `sudo` для повышения привилегий
- `audio` - управление аудиодрайвером
- `src` - полный доступ к исходникам в каталоге `/usr/src/`
- `shadow` - разрешает чтение файла `/etc/shadow`
- `utmp` - разрешает запись в файлы `/var/log/utmp` `/var/log/wtmp`
- `video` - позволяет работать с видеодрайвером
- `plugdev` - позволяет монтировать внешние устройства USB, CD и т.д.
- `staff` - разрешает запись в папку `/usr/local`

4 Выполнение лабораторной работы

5 Цель работы

Получение практических навыков работы в консоли с расширенными атрибутами файлов

6 Теоретическое введение

Права доступа определяют, какие действия конкретный пользователь может или не может совершать с определенными файлами и каталогами. С помощью разрешений можно создать надежную среду — такую, в которой никто не может поменять содержимое ваших документов или повредить системные файлы. [1]

Расширенные атрибуты файлов Linux представляют собой пары имя:значение, которые постоянно связаны с файлами и каталогами, подобно тому как строки окружения связаны с процессом. Атрибут может быть определён или не определён. Если он определён, то его значение может быть или пустым, или не пустым. [2]

Расширенные атрибуты дополняют обычные атрибуты, которые связаны со всеми inode в файловой системе (т. е., данные stat(2)). Часто они используются для предоставления дополнительных возможностей файловой системы, например, дополнительные возможности безопасности, такие как списки контроля доступа (ACL), могут быть реализованы через расширенные атрибуты. [3]

Установить атрибуты:

- `chattr filename`

Значения:

- `chattr +a #` только добавление. Удаление и переименование запрещено;
- `chattr +A #` не фиксировать данные об обращении к файлу
- `chattr +c #` сжатый файл

- `chattr +d` # неархивируемый файл
- `chattr +i` # неизменяемый файл
- `chattr +S` # синхронное обновление
- `chattr +s` # безопасное удаление, (после удаления место на диске переписывается нулями)
- `chattr +u` # неудаляемый файл
- `chattr -R` # рекурсия

Просмотреть атрибуты:

- `lsattr filename`

Опции:

- `lsattr -R` # рекурсия
- `lsattr -a` # вывести все файлы (включая скрытые)
- `lsattr -d` # не выводить содержимое директории

7 Выполнение лабораторной работы

1. От имени пользователя guest, созданного в прошлых лабораторных работах, определяю расширенные атрибуты файла /home/guest/dir1/file1 (рис. 1).

```
[ivmurashov@ivmurashov ~]$ sudo useradd guest2
[sudo] password for ivmurashov:
[ivmurashov@ivmurashov ~]$ sudo passwd guest2
Changing password for user guest2.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
```

Рис. 7.1: Определение атрибутов

2. Изменяю права доступа для файла home/guest/dir1/file1 с помощью chmod 600 (рис. 2).

```
[ivmurashov@ivmurashov ~]$ sudo gpasswd -a guest2 guest
Adding user guest2 to group guest
```

Рис. 7.2: Изменение прав доступа

3. Пробую установить на файл /home/guest/dir1/file1 расширенный атрибут а от имени пользователя guest, в ответ получаю отказ от выполнения операции (рис. 3).

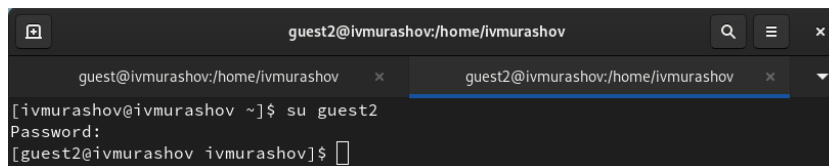


Рис. 7.3: Попытка установки расширенных атрибутов

4. Устанавливаю расширенные права уже от имени суперпользователя, теперь нет отказа от выполнения операции (рис. 4).

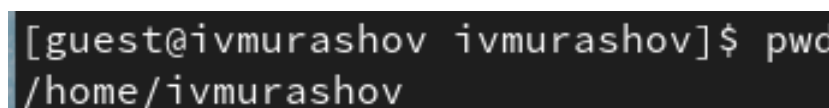


Рис. 7.4: Установка расширенных атрибутов

5. От пользователя guest проверяю правильность установки атрибута (рис. 5).

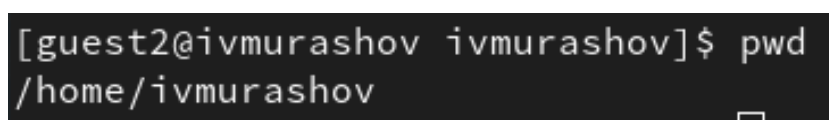


Рис. 7.5: Проверка атрибутов

6. Выполняю **дозапись** в файл с помощью `echo 'test' >> dir1/file1`, далее выполняю чтение файла, убеждаюсь, что дозапись была выполнена (рис. 6).

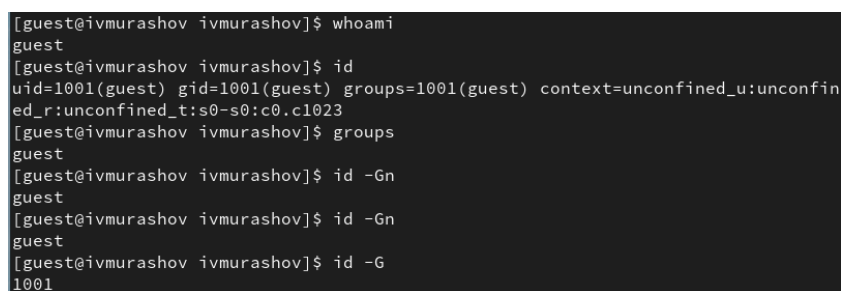


Рис. 7.6: Дозапись в файл

7. Пробую удалить файл, получаю отказ от выполнения действия. (рис. 7).

```
[guest2@ivmurashov ivmurashov]$ whoami
guest2
[guest2@ivmurashov ivmurashov]$ id
uid=1002(guest2) gid=1002(guest2) groups=1002(guest2),1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest2@ivmurashov ivmurashov]$ groups
guest2 guest
[guest2@ivmurashov ivmurashov]$ id -Gn
guest2 guest
[guest2@ivmurashov ivmurashov]$ id -G
1002 1001
```

Рис. 7.7: Попытка удалить файл

То же самое получаю при попытке переименовать файл(рис. 8).

```
[guest2@ivmurashov ivmurashov]$ cat /etc/group | grep 'guest'
guest:x:1001:guest2
guest2:x:1002:
```

Рис. 7.8: Попытка переименовать файл

8. Получаю отказ от выполнения при попытке установить другие права доступа (рис. 9).

```
[guest2@ivmurashov ivmurashov]$ newgrp guest
[guest2@ivmurashov ivmurashov]$
```

Рис. 7.9: Попытка изменить права доступа

9. Снимаю расширенные атрибуты с файла (рис. 10).

```
[guest@ivmurashov ivmurashov]$ cd
[guest@ivmurashov ~]$ pwd
/home/guest
[guest@ivmurashov ~]$ chmod g+rxw /home/guest
```

Рис. 7.10: Снятие расширенных атрибутов

Проверяю ранее не удавшиеся действия: чтение, переименование, изменение прав доступа. Теперь все из этого выполняется (рис. 11).

```
[guest@ivmurashov ~]$ chmod 000 dir1
[guest@ivmurashov ~]$ ls
Desktop dir1 Documents Downloads Music Pictures Public Templates test Videos
[guest@ivmurashov ~]$ ls -l
total 4
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Desktop
d----- 2 guest guest 6 Mar 7 13:14 dir1
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Documents
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Downloads
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Music
drwxr-xr-x. 2 guest guest 6 Mar 7 15:19 Pictures
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Public
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Templates
-rw-r--r--. 1 guest guest 5 Mar 7 13:10 test
drwxr-xr-x. 2 guest guest 6 Mar 7 12:53 Videos
```

Рис. 7.11: Проверка выполнения действий

10. Пытаюсь добавить расширенный атрибут `i` от имени пользователя `guest`, как и раньше, получаю отказ (рис. 12).

```
[guest2@ivmurashov ~]$ cd /home/guest
[guest2@ivmurashov guest]$ ls
Desktop dir1 Documents Downloads Music Pictures Public Templates test Videos
[guest2@ivmurashov guest]$ ls dir1
ls: cannot open directory 'dir1': Permission denied
[guest2@ivmurashov guest]$ rm dir1/a
rm: cannot remove 'dir1/a': Permission denied
[guest2@ivmurashov guest]$ touch dir1/s
touch: cannot touch 'dir1/s': Permission denied
[guest2@ivmurashov guest]$ echo 'test' > dir1/file1
bash: dir1/file1: Permission denied
[guest2@ivmurashov guest]$ cat dir1/file1
cat: dir1/file1: Permission denied
[guest2@ivmurashov guest]$ chmod 020 dir1/file1
chmod: cannot access 'dir1/file1': Permission denied
```

Рис. 7.12: Попытка добавить расширенный атрибут

Добавляю расширенный атрибут `i` от имени суперпользователя, теперь все выполнено верно (рис. 13).

```
[guest@ivmurashov ivmurashov]$ whoami
guest
[guest@ivmurashov ivmurashov]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@ivmurashov ivmurashov]$ groups
guest
[guest@ivmurashov ivmurashov]$ id -Gn
guest
[guest@ivmurashov ivmurashov]$ id -Gn
guest
[guest@ivmurashov ivmurashov]$ id -G
1001
```

Рис. 7.13: Добавление расширенного атрибута

Пытаюсь записать в файл, дозаписать, переименовать или удалить, ничего из этого сделать нельзя (рис. 14).


```
[guest2@ivmurashov ivmurashov]$ whoami
guest2
[guest2@ivmurashov ivmurashov]$ id
uid=1002(guest2) gid=1002(guest2) groups=1002(guest2),1001(guest) context=unconfined_u:un
confined_r:unconfined_t:s0-s0:c0.c1023
[guest2@ivmurashov ivmurashov]$ groups
guest2 guest
[guest2@ivmurashov ivmurashov]$ id -Gn
guest2 guest
[guest2@ivmurashov ivmurashov]$ id -G
1002 1001
```

Рис. 7.14: Проверка выполнения действий

8 Выводы

В результате выполнения работы вы 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»