Отчёт по лабораторной работе №5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Предмет: информационная безопасность

Александр Сергеевич Баклашов

Содержание

[1 Цель работы 1](#_Toc116155877)

[2 Теоретическое введение 1](#_Toc116155878)

[3 Выполнение лабораторной работы 2](#_Toc116155879)

[3.1 Создание программы 2](#_Toc116155880)

[3.2 Исследование Sticky-бита 7](#_Toc116155881)

[4 Вывод 10](#_Toc116155882)

[5 Библиография 10](#_Toc116155883)

# 1 Цель работы

Изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов. Получение практических навыков работы в консоли с дополнительными атрибутами. Рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление 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

## 3.1 Создание программы

1. Войдите в систему от имени пользователя guest. (рис. [1](#fig:001))

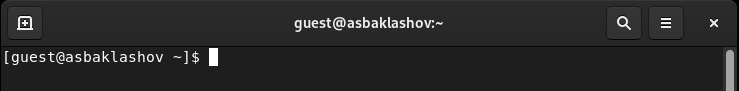


Figure 1: guest

1. Создайте программу simpleid.c: (рис. [2](#fig:002))

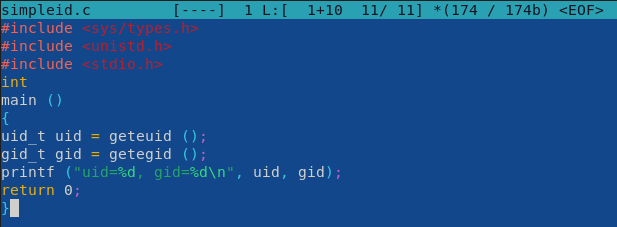


Figure 2: simpleid.c

1. Скомплилируйте программу и убедитесь, что файл программы создан (рис. [3](#fig:003))

Figure 3: Скомплилируем

Figure 3: Скомплилируем

1. Выполните программу simpleid

Выполните системную программу id и сравните полученный вами результат с данными предыдущего пункта задания. (рис. [4](#fig:004))

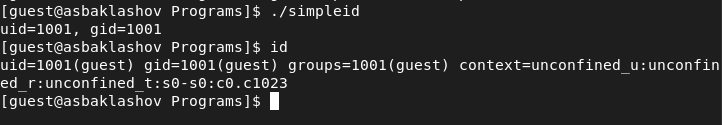


Figure 4: id

Результаты совпадают.

1. Усложните программу, добавив вывод действительных идентификато- ров. (рис. [5](#fig:005))

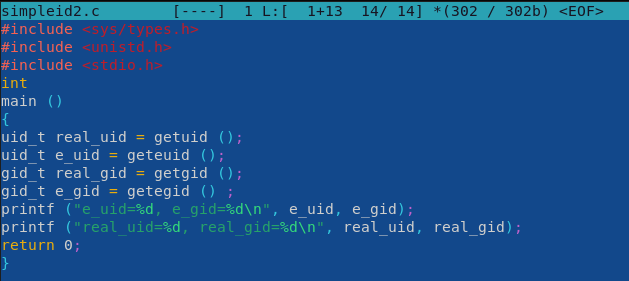


Figure 5: Усложним программу

Получившуюся программу назовите simpleid2.c

1. Скомпилируйте и запустите simpleid2.c (рис. [6](#fig:006))

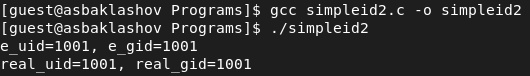


Figure 6: simpleid2.c

1. От имени суперпользователя выполните команды.

chown root:guest /home/guest/simpleid2

chmod u+s /home/guest/simpleid2 (рис. [7](#fig:007))

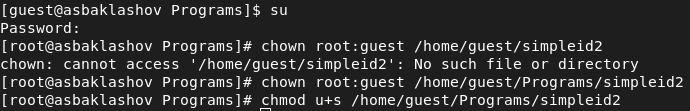


Figure 7: chmod

Используйте sudo или повысьте временно свои права с помощью su. Поясните, что делают эти команды.

1ая команда меняет владельца, 2ая - атрибуты

1. Выполните проверку правильности установки новых атрибутов и смены владельца файла simpleid2 (рис. [8](#fig:008))

Figure 8: ls

Figure 8: ls

1. Запустите simpleid2 и id (рис. [9](#fig:009))

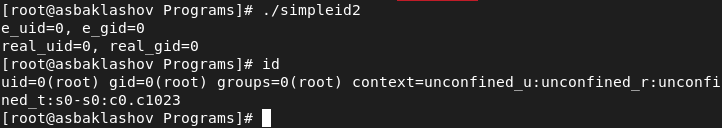


Figure 9: id

Результаты совпадают.

1. Проделайте тоже самое относительно SetGID-бита (рис. [10](#fig:010))

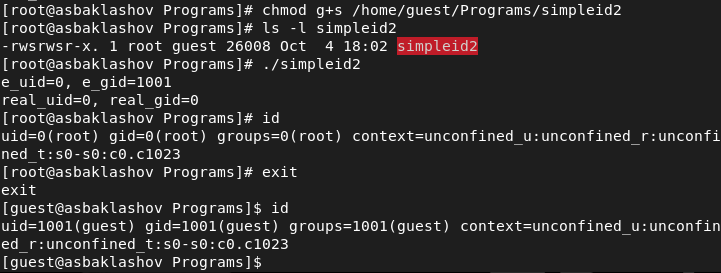


Figure 10: SetGID

1. Создайте программу readfile.c (рис. [11](#fig:011))



Figure 11: readfile.c

1. Откомпилируйте её (рис. [12](#fig:012))

Figure 12: Compile

Figure 12: Compile

1. Смените владельца у файла readfile.c (или любого другого текстового файла в системе) и измените права так, чтобы только суперпользователь (root) мог прочитать его, a guest не мог (рис. [13](#fig:013))

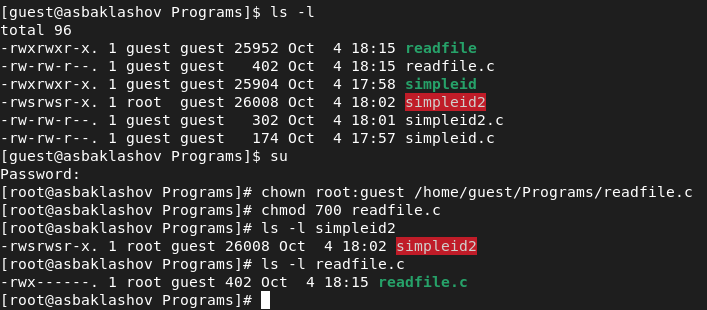


Figure 13: Смена владельца

1. Проверьте, что пользователь guest не может прочитать файл readfile.c. (рис. [14](#fig:014))

Figure 14: Проверка

Figure 14: Проверка

1. Смените у программы readfile владельца и установите SetU’D-бит (рис. [15](#fig:015))

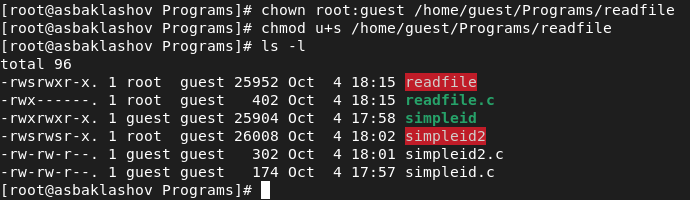


Figure 15: Смена владельца

1. Проверьте, может ли программа readfile прочитать файл readfile.c? (рис. [16](#fig:016))

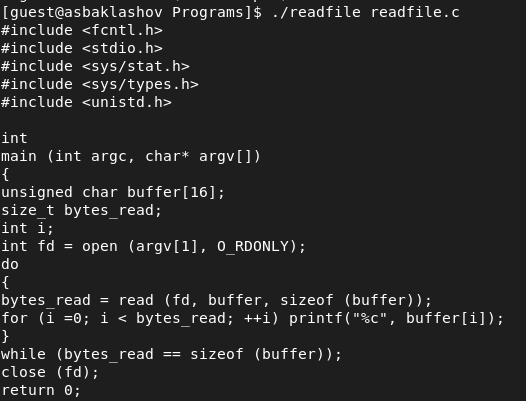


Figure 16: Проверка

1. Проверьте, может ли программа readfile прочитать файл /etc/shadow? (рис. [17](#fig:017))



Figure 17: Проверка

Программа может прочитать оба файла.

## 3.2 Исследование Sticky-бита

1. Выясните, установлен ли атрибут Sticky на директории /tmp, для чего выполните команду ls -l / | grep tmp (рис. [18](#fig:018))

Figure 18: Sticky

Figure 18: Sticky

Атрибут “t” установлен.

1. От имени пользователя guest создайте файл file01.txt в директории /tmp со словом test

Просмотрите атрибуты у только что созданного файла и разрешите чтение и запись для категории пользователей «все остальные». (рис. [19](#fig:019))

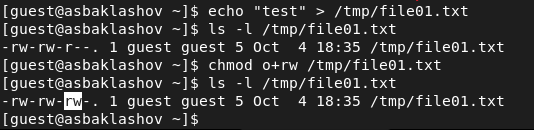


Figure 19: file01.txt

1. От пользователя guest2 (не являющегося владельцем) попробуйте прочитать файл /tmp/file01.txt

От пользователя guest2 попробуйте дозаписать в файл /tmp/file01.txt слово test2 командой

Проверьте содержимое файла командой cat /tmp/file01.txt (рис. [20](#fig:020))

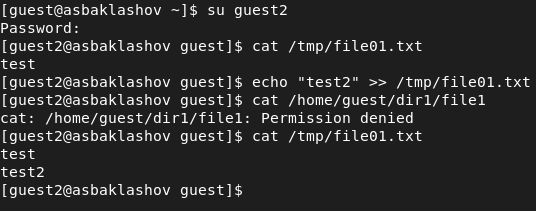


Figure 20: атрибуты

1. От пользователя guest2 попробуйте записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию командой

Проверьте содержимое файла командой cat /tmp/file01.txt (рис. [21](#fig:021))

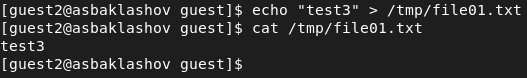


Figure 21: запись в файл

1. От пользователя guest2 попробуйте удалить файл /tmp/file01.txt командой rm /tmp/fileOl.txt (рис. [22](#fig:022))

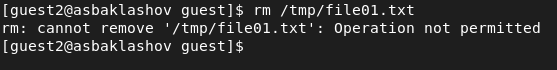


Figure 22: удаление файла

Удалить файл не удалось

1. Повысьте свои права до суперпользователя следующей командой su - и выполните после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp: chmod -t /tmp

Покиньте режим суперпользователя командой exit

От пользователя guest2 проверьте, что атрибута t у директории /tmp нет: ls -l / | grep tmp (рис. [23](#fig:023))

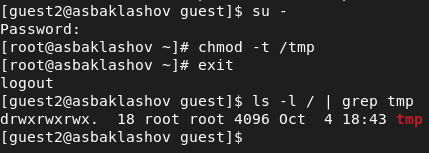


Figure 23: Снятие Sticky-бита

1. Повторим предыдущие шаги (рис. [24](#fig:024), рис. [25](#fig:025))

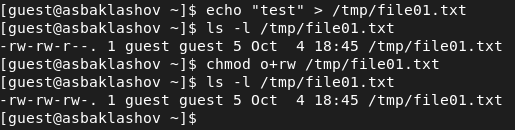


Figure 24: шаги

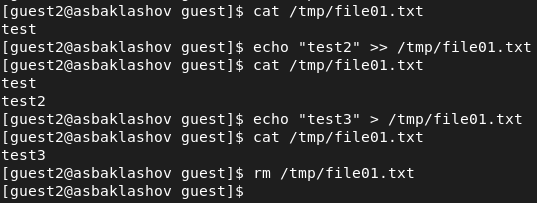


Figure 25: шаги

1. Повысьте свои права до суперпользователя и верните атрибут t на директорию /tmp (рис. [26](#fig:026))

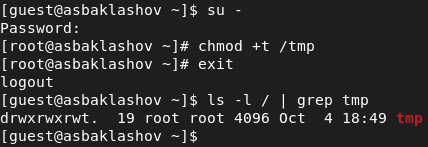


Figure 26: Возвращение Sticky-бита

# 4 Вывод

В результате выполнения работы я изучил механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# 5 Библиография

1. Лабораторная работа №5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов. - 7 с. [Электронный ресурс]. М. URL: [Лабораторная работа №5](https://esystem.rudn.ru/pluginfile.php/1651889/mod_resource/content/2/005-lab_discret_sticky.pdf) (Дата обращения: 08.10.2022).
2. Rocky Linux Documentation. [Электронный ресурс]. М. URL: [Rocky Linux Documentation](https://docs.rockylinux.org) (Дата обращения: 08.10.2022).
3. Chmod. [Электронный ресурс]. М. URL: [Chmod](https://ru.wikipedia.org/wiki/Chmod) (Дата обращения: 08.10.2022).