Лабораторная работа № 5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Абу Сувейлим Мухаммед Мунифович

Содержание

# 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.

# 2 Задание

Создание программы и исследование Sticky-бита

# 3 Теоретическое введение

В метаданных каждого объекта (в inode файла/директории) содержится список разрешений на доступ к нему для разных категорий субъектов [1].

Атрибуты Minimal ACL поддерживают три базовых класса субъектов доступа к файлу (класс All объединяет все три класса):

User access (u) – доступ для владельца файла; Group access (g) – доступ для группы, владеющей файлом; Other access (o) – доступ для остальных пользователей (кроме пользователя root). All access (a) – доступ для всех субъектов доступа (u, g, o). Для каждого из этих классов определены три типа разрешений:

На чтение содержимого файла (read) – символ «r». На запись внутри файла или изменения его содержимого (write) – символ «w». На исполнение файла (если это бинарный исполняемый файл или файл сценария интерпретатора (execute)) – символ «x» [2].

# 4 Выполнение лабораторной работы

Проверяем, что у нас установлен компилятор gcc командой (рис. 1):

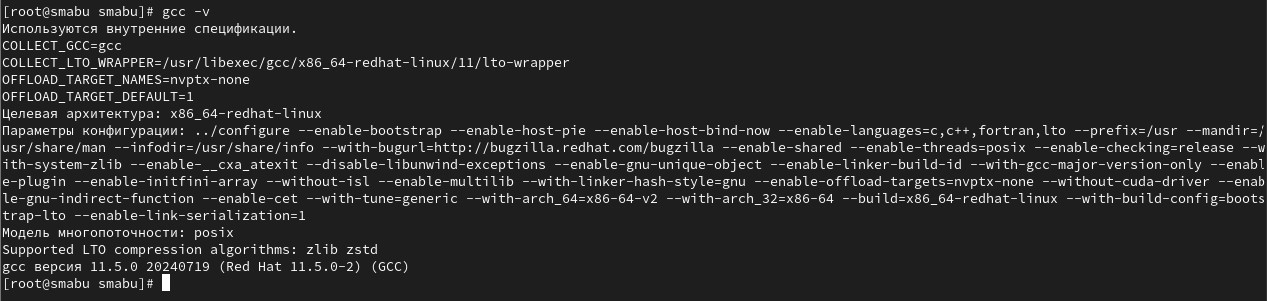


Рис. 1: Компилятор gcc

Далее, отключим систему запретов до очередной перезагрузки системы командой (рис. 2):

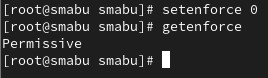


Рис. 2: Команда setenforce

Войдем в систему от имени пользователя guest и создадим программу simpleid.c (рис. 3):



Рис. 3: Программа simpleid.c

Скомплилируйем программу и убедимся, что файл программы создан (рис. 4):



Рис. 4: Запуск программы simpleid.c

Выполним программу simpleid и системную программу id. Видем, что полученный нами результат с данными предыдущего пункта задания и этого пункта задания совпадают.

Усложним программу, добавив вывод действительных идентификаторов (рис. 5):

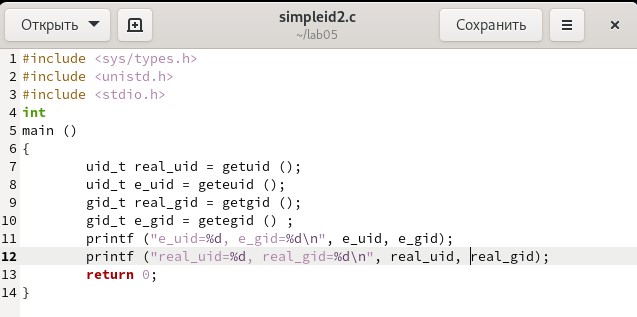


Рис. 5: Программа simpleid2.c

Скомпилируйем и запустим simpleid2.c (рис. 6):

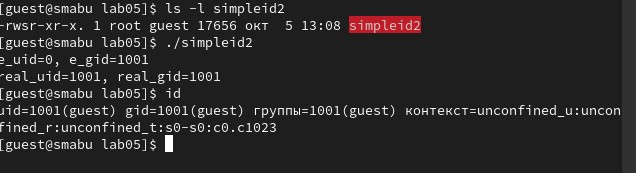


Рис. 6: Запуск программы simpleid2.c

От имени суперпользователя выполним команды:

chown root:guest /home/guest/simpleid2  
chmod u+s /home/guest/simpleid2

Выполним проверку правильности установки новых атрибутов и смены владельца файла simpleid2 и запустим simpleid2 и id (рис. 7):

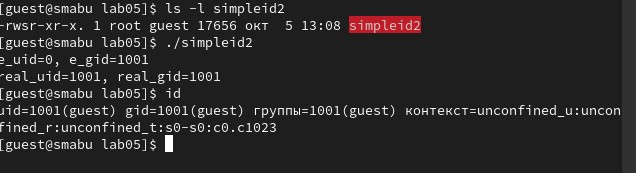


Рис. 7: Проверка правильности установки новых атрибутов

Создаим программу readfile.c (рис. 8):

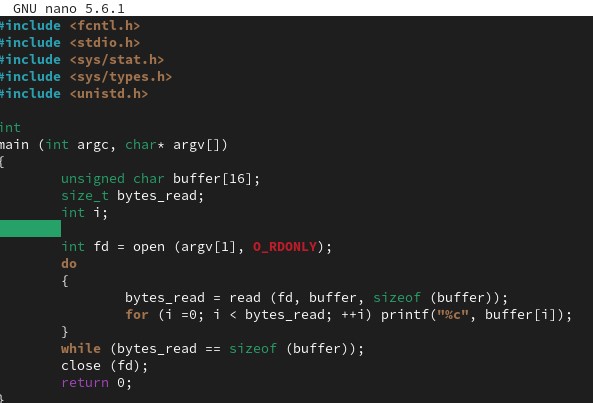


Рис. 8: Программа readfile.c

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. 9):

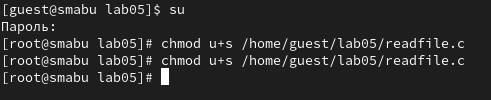


Рис. 9: Смена владельца у файла readfile.c

Программа readfile неможет прочитать файл readfile.c и неможет ипрочитать файл /etc/shadow (рис. 10)

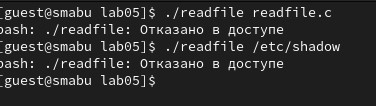


Рис. 10: Чтения файла readfile.c

Выясним, установлен ли атрибут Sticky на директории /tmp, для чего выполним команду (рис. 11):

Команда grep

Рис. 11: Команда grep

От имени пользователя guest создадим файл file01.txt в директории /tmp со словом test (рис. 12):

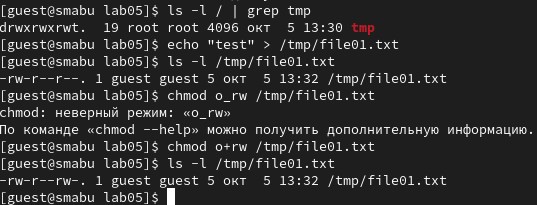


Рис. 12: Файл file01.txt

Просмотрим атрибуты у только что созданного файла и разрешите чтение и запись для категории пользователей «все остальные» (рис. 12).

От пользователя guest2 (не являющегося владельцем) попробуйм прочитать файл /tmp/file01.txt (рис. 13):

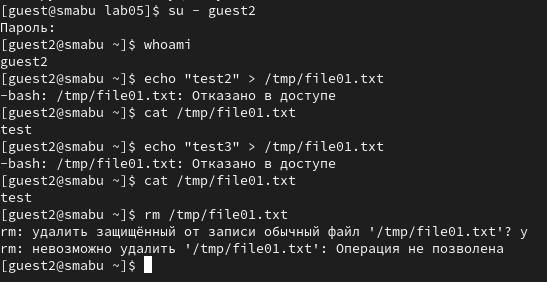


Рис. 13: Чтения файла file01.txt от пользователя guest2

Попробуйм дозаписать в файл /tmp/file01.txt слово test2 командой (рис. 13). Проверем содержимое файла командой (рис. 13). От пользователя guest2 попробуйм записать в файл /tmp/file01.txt слово test3, стерев при этом всю имеющуюся в файле информацию командой (рис. 13). Проверем содержимое файла и попробуйм удалить файл /tmp/file01.txt командой (рис. 13).

Выполним после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp (рис. 14):

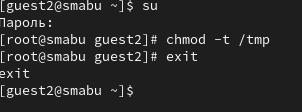


Рис. 14: Команда chmod -t /tmp

От пользователя guest2 проверем, что атрибута t у директории /tmp нет (рис. 15):

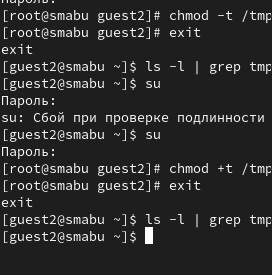


Рис. 15: Команда chmod -t /tmp

# 5 Выводы

Изучали механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами. Рассмотрели работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов кратко описываются итоги проделанной работы.

# Список литературы

1. [// skillbox.ru](https://skillbox.ru/media/code/kali-linux-obzor-distributiva-dlya-budushchikh-khakerov/).

2. Таненбаум Э., Бос Х. Современные операционные системы. 4-е изд. СПб.: Питер, 2015. 1120 с.