Отчет по лабораторной работе №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Галацан Николай, НПИбд-01-22

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. [1]

# 2 Выполнение лабораторной работы [1]

## 2.1 Создание программы

От имени пользователя guest создаю программу simpleid.c (рис. 1).



Рис. 1: Программа simpleid.c

Компилирую программу командой gcc simpleid.c -o simpleid и выполняю. Выполняю системную программу id и убеждаюсь, что обе программы выводят одинаковые данные (рис. 2).

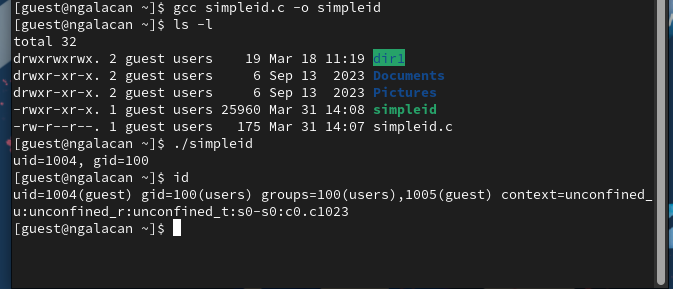


Рис. 2: Компиляция и выполнение. Сравнение с id

Cоздаю усложненную программу simpleid2.c (рис. 3).

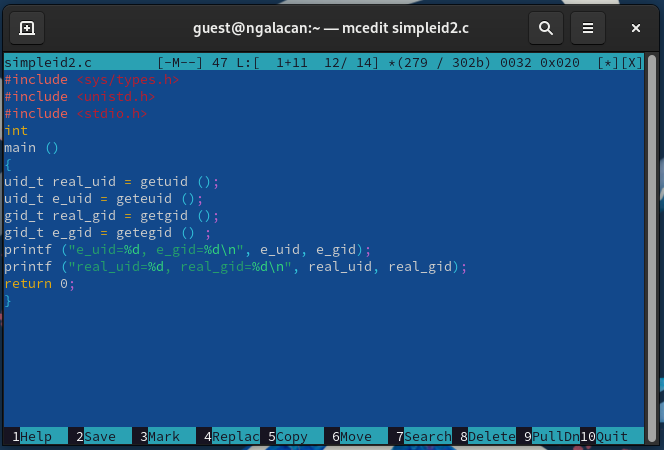


Рис. 3: ОПрограмма simpleid2.c

Компилирую и запускаю (рис. 4).

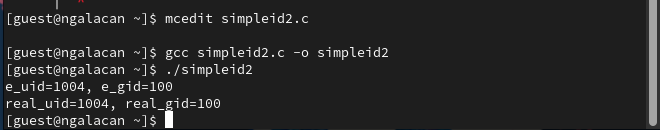


Рис. 4: Компиляция и выполнение

От имени суперпользователя изменяю владельца файла и добавляю атрибут s. Это означает, что пользователь будет выполнять файл с разрешениями владельца файла. Проверяю правильность и запускаю программу, вновь сравниваю с id. Исходя из этого, можно сказать, что теперь владельцем файла является пользователь с id 0 (root), а изначально владельцем файла был пользователь с id 1004 (guest) (рис. 5).

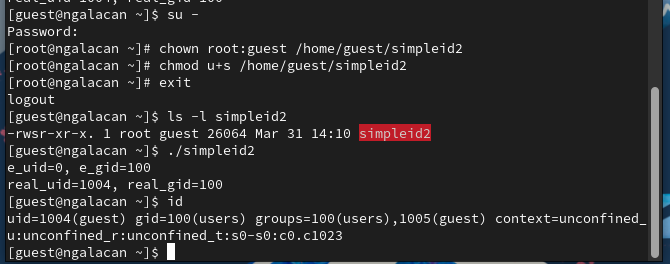


Рис. 5: Запуск simpleid2 с SetUID. Сравнение результатов

То же самое проделываю для SetGID бита (рис. 6).

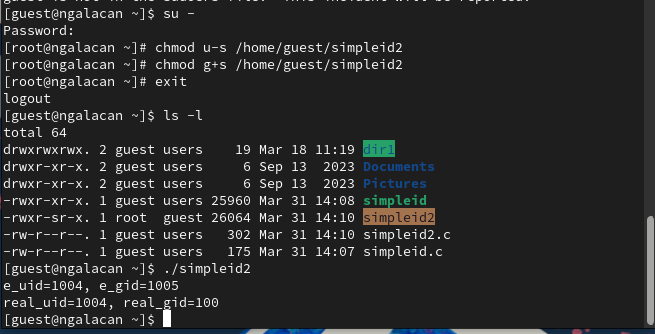


Рис. 6: Запуск simpleid2 с SetGID. Сравнение результатов

Создаю программу readfile.c (рис. 7).

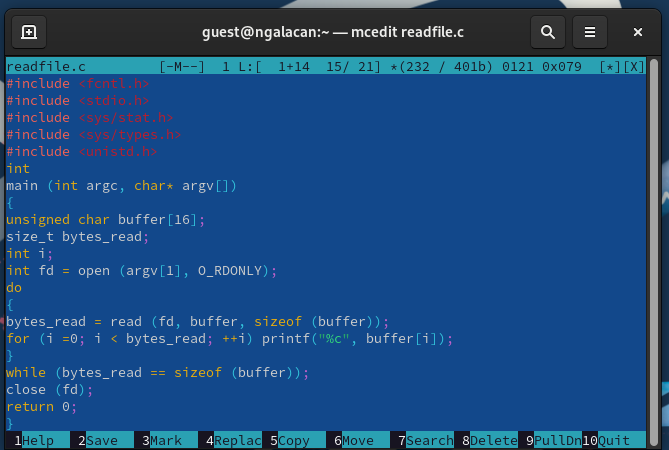


Рис. 7: Программа readfile.c

Компилирую ее, изменяю у файла readfile.c владельца, изменяю права доступа так, чтобы только суперпользователь мог прочитать его, a guest не мог. После проверяю, что пользователь guest не может прочитать файл (рис. 8).



Рис. 8: Изменение прав доступа, проверка от имени пользователя guest

Изменяю у программы readfile владельца, устанавливаю SetUID-бит. Проверяю, что программа может прочитать файлы readfile.c и /etc/shadow (рис. 9).

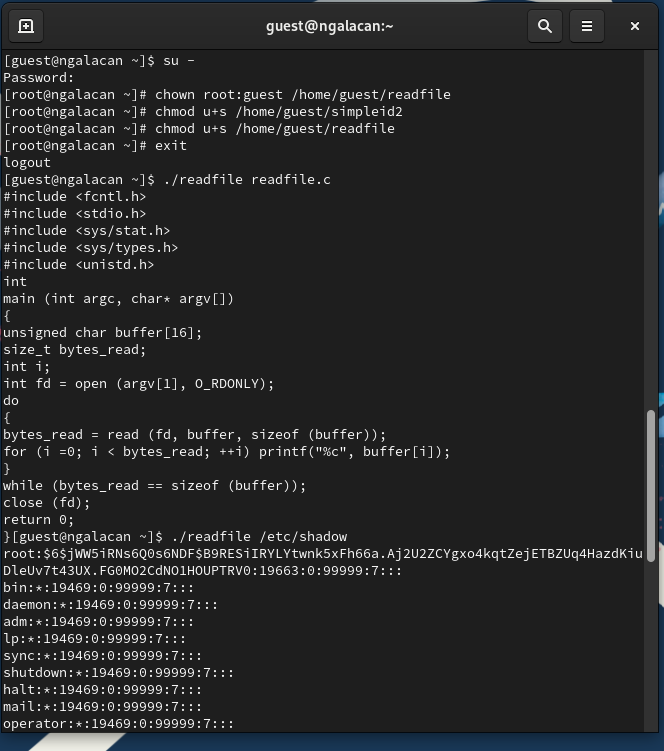


Рис. 9: Установка SetUID для readfile и проверка

## 2.2 Исследование Sticky-бита

Проверяю, установлен ли атрибут Sticky на директории /tmp. От пользователя guest создаю файл со словом test командой echo “test” > /tmp/file01.txt. Просматриваю атрибуты у только что созданного файла и разрешаю чтение и запись для остальных пользователей. От пользователя guest2 пробую прочитать файл (успешно) и внести изменения в файл (отказ в доступе) (рис. 10).

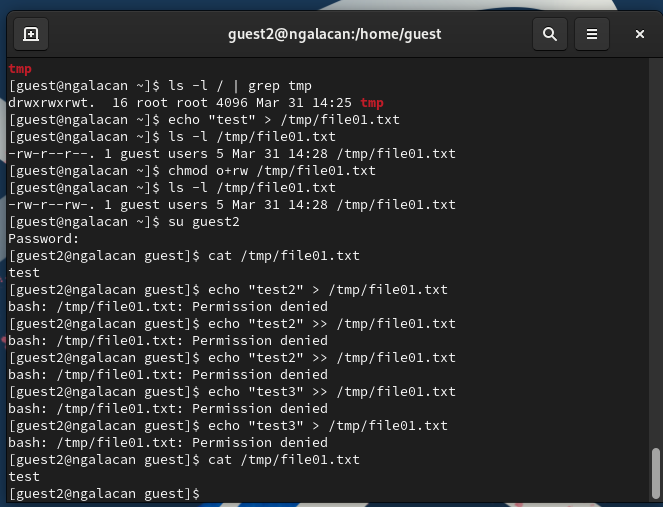


Рис. 10: Создание файла, изменение прав, просмотр и попытки записи

Пытаюсь удалить файл (отказ в доступе) (рис. 11).

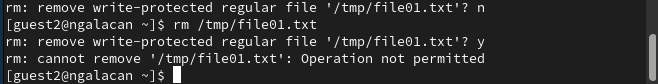


Рис. 11: Попытка удаления

От имени суперпользователя снимаю атрибут t с директории /tmp и от имени guest2 проверяю. Повторяю предыдущие шаги: просмотр файла разрешен, запись в файл не разрешена, удаление файла разрешено. В конце возвращаю атрибут t на директорию /tmp от имени суперпользователя (рис. 12).

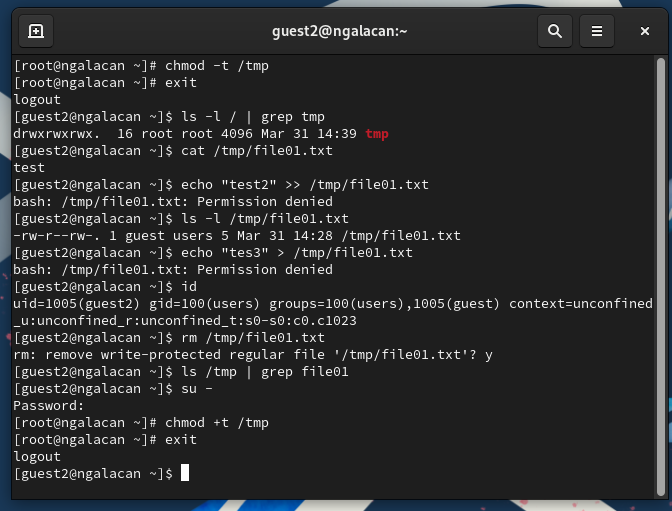


Рис. 12: Повторение операций без атрибута t

# 3 Выводы

Были изучены механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получены практические навыки работы в консоли с дополнительными атрибутами. Рассмотрены работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Список литературы

1. Кулябов Д.С., Королькова А.В., Геворкян М.Н. Информационная безопасность компьютерных сетей. Лабораторные работы, учебное пособие. Москва: РУДН, 2015. 64 с.