Отчёт по лабораторной работе

Лабораторная №5

Дерябина Мария Сергеевна

Содержание

# Цель работы

Изучить механизмы изменения идентификаторов, применения SetUID- и Sticky-биты. Получить практические навыки работы в консоли с дополнительными атрибутами. Рассмотреть работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Выполнение лабораторной работы

Для выполнения работы, установила компилятор gcc и отключила защиту SELinux (рис. 1, 2).

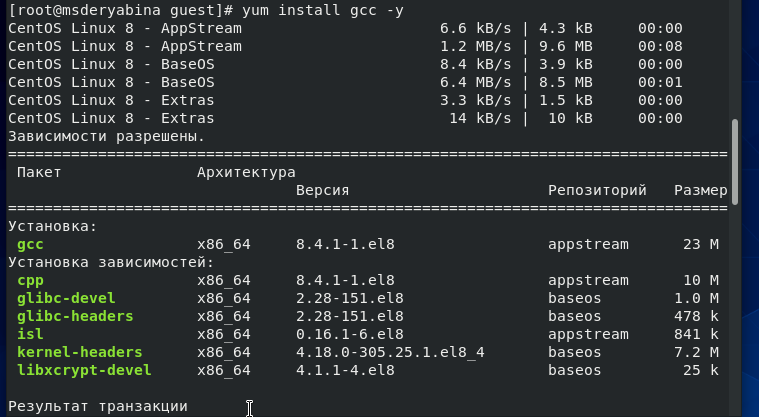


Figure 1: Установка gcc

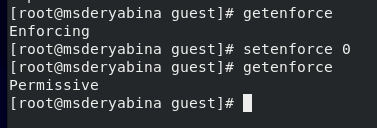


Figure 2: Снятие ограничений SELinux

## Исследование SetUID- и SetGID-битов

Вошла в систему от пользователя guest и создала программу simpleid.c (рис. 3).

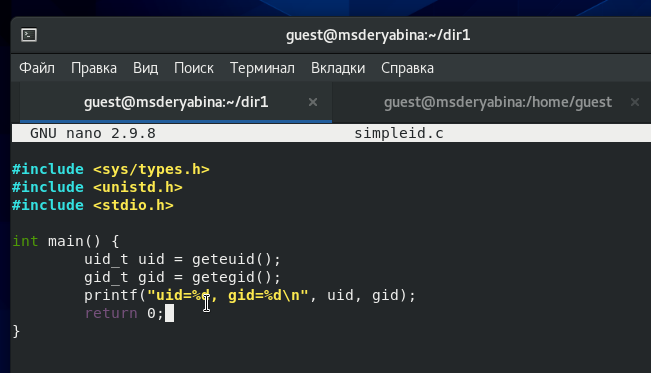


Figure 3: Код программы simpleid.c

Скомпилировала и выполнила программу. Полученный результат совпал с выводом команды id (рис. 4)

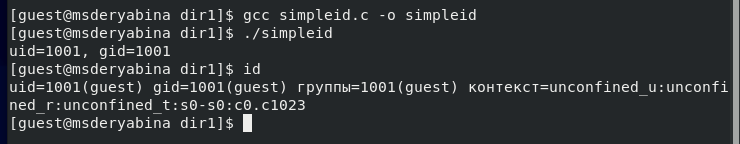


Figure 4: Компиляция и выполнение программы simpleid.c

Добавила в программу вывод действительных идентификаторов, назвала ее simpleid2.c (рис. 5).

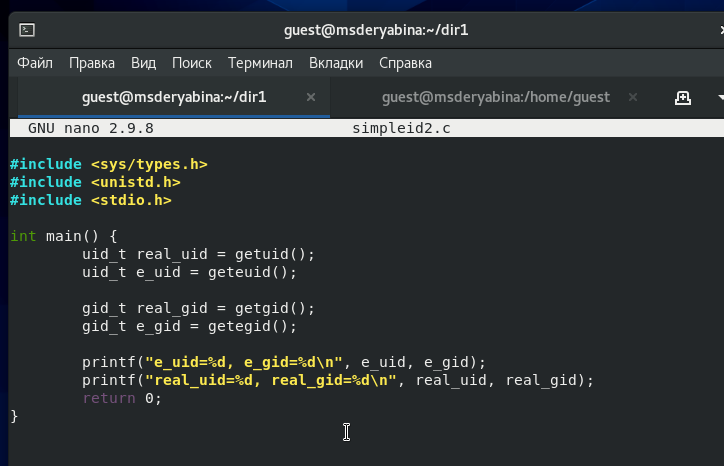


Figure 5: Код программы simpleid2.c

Скомпилировала и запустила программу simpleid2.c. Действительные идентификаторы совпали с эффективными (рис. 6)

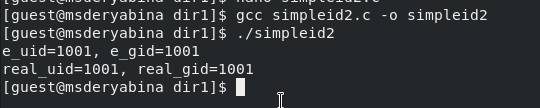


Figure 6: Компиляция и выполнение программы simpleid2.c

От имени суперпользователя изменила владельца программы simpleid2 на root и добавила атрибут SetUID. (рис. 7)

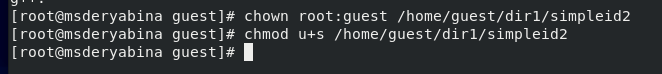


Figure 7: Изменение атрибутов программы simpleid2

Проверила правильность установки новых атрибутов и смены владельца файла simpleid2 и запустила simpleid2. Теперь вывод программы отличается от вывода команды id. Действительные идентификаторы остались прежними, а эффективный идентификатор пользователя теперь равен 0 - это идентификатор суперпользователя. Это значит, что пользователь guest использует права суперпользователя во время выполнения программы (рис. 8)

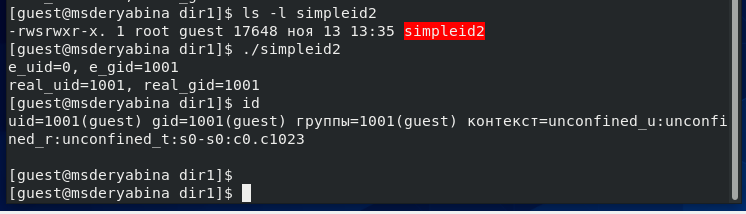


Figure 8: Вывод программы simpleid2 с атрибутом SetUID

Проделала то же самое относительно SetGID-бита. Результат оказался аналогичным, теперь при выполнении simpleid2 от пользователя guest эффектиынй идентификатор группы равени идентификатору группы суперпользователя (рис. 9, 10)

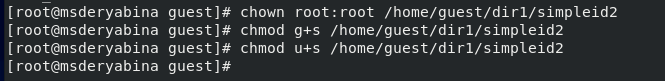


Figure 9: Добавление атрибута SetGID к программе simpleid2

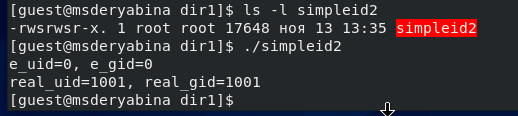


Figure 10: Вывод программы simpleid2 с атрибутом SetUID и SetGID

Создала программу readfile.c (рис. 11)

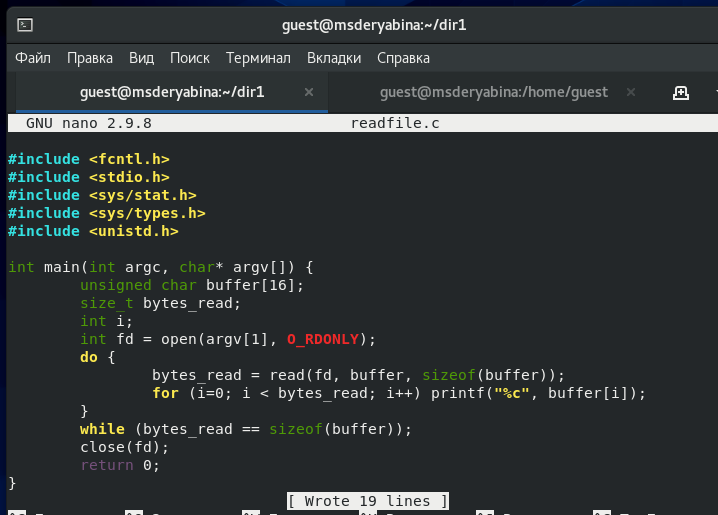


Figure 11: Код программы readfile.c

Откомпилировала и проверила корректность выполения программы (рис. 12)

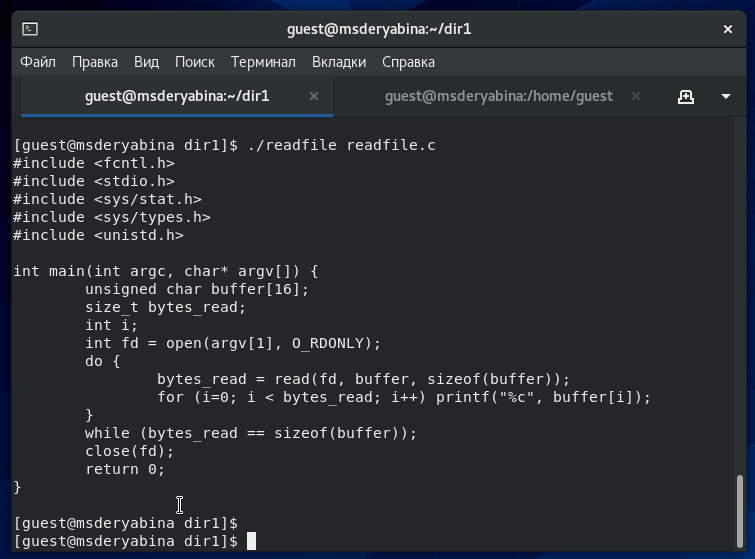


Figure 12: Выполнение программы readfile

Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь мог прочитать его, а guest не мог (рис. 13)

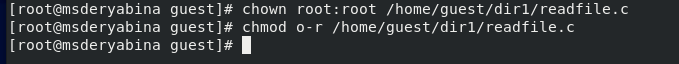


Figure 13: Смена атрибутов файла readfile.c

Проверила, что пользователь guest не может прочитать файл readfile.c (рис. 14)

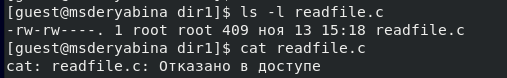


Figure 14: Проверка атрибутов файла readfile.c

Сменила у программы readfile вдадельца на root и установила SetUID-бит (рис. 15)

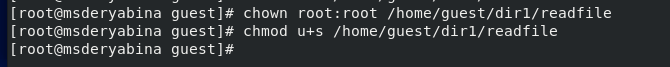


Figure 15: Добавление SetUID-бита к программе readfile

Теперь с помощью программы readfile можно от имени пользователя guest прочитать файл readfile.c. Также можно прочитать файл /etc/shadow, хотя guest не имеет к нему доступа. (рис. 16, 17)

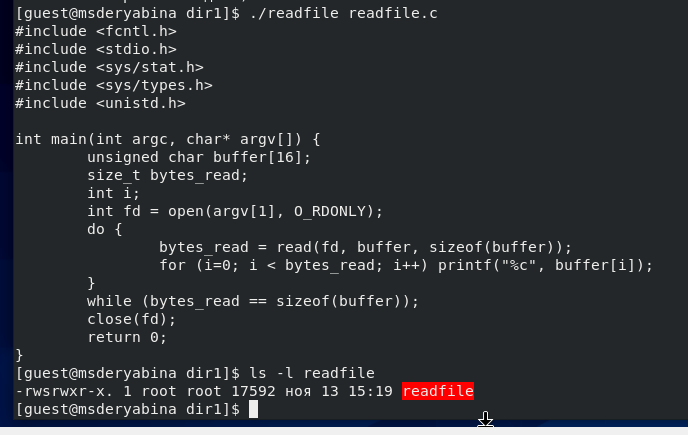


Figure 16: Чтение файла readfile.c с помощью readfile

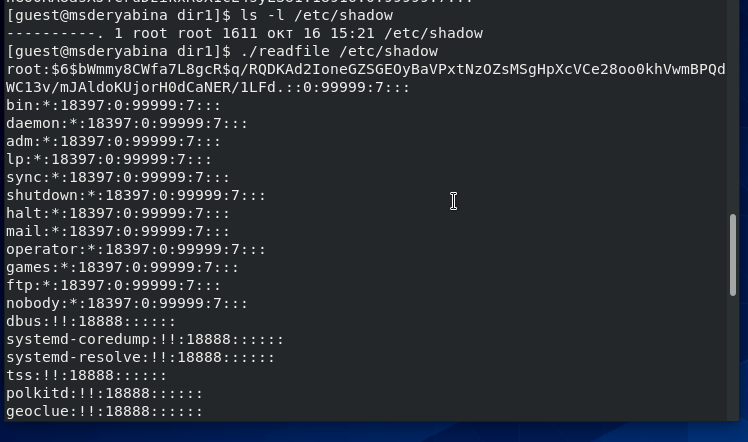


Figure 17: Чтение файла /etc/shadow с помощью readfile

## Исследование Sticky-бита

Посмотрела, что на директории /tmp установлен атрибут Sticky. От имени пользователя guest создала файл file01.txt в директории /tmp со словом “test”. Посмотрела атрибуты у file01.txt и разрешила чтение и запись для категории пользователей “other” (рис. 18)

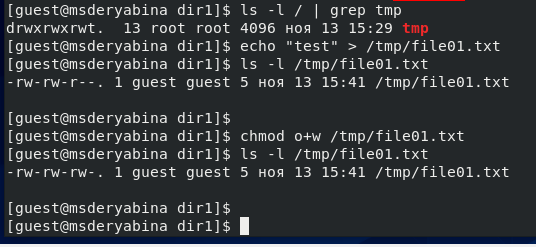


Figure 18: Проверка атрибута Sticky и создание файла в /tmp

От пользователя guest2 попробовала выполнить различные действия - прочитать файл, дозаписать текст в файл, переписать текст в файле, удалить файл. Получилось сделать все, кроме удаления файла (рис. 19)

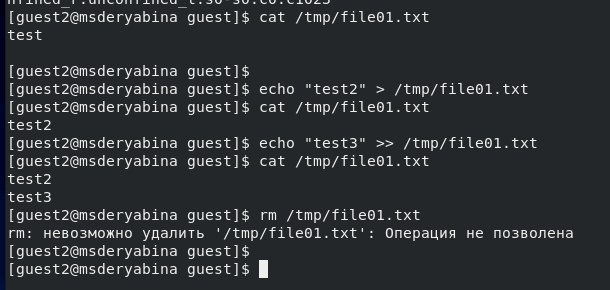


Figure 19: Выполнение операций над file01.txt от имени guest2

От имени суперпользователя сняла Sticky-бит с директории /tmp (рис. 20)

Figure 20: Снятие атрибута Sticky с /tmp

Figure 20: Снятие атрибута Sticky с /tmp

Повторила предыдущие шаги. В этот раз удалось удалить file01.txt.

Таким образом, со снятым атрибутом Sticky можно удалить из директории файл от имени пользователя, не являющегося его владельцем. Вернула атрибут t на директорию /tmp (рис. 21)

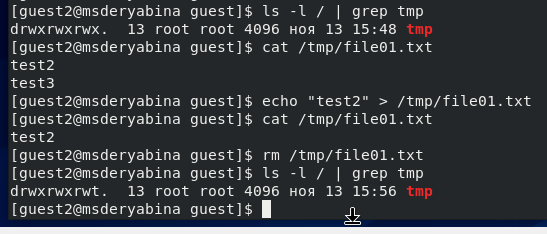


Figure 21: Выполнение операций над file01.txt со снятым атрибутом Sticky

# Вывод

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-биты. Получила практические навыки работы в консоли с дополнительными атрибутами. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.