Лабораторная работа No 5.

Тагиев Байрам Алтай оглы

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 Выполнение лабораторной работы

## 2.1 SetUID бит

1. Из-под пользователя guest создадим файл simpleid.c

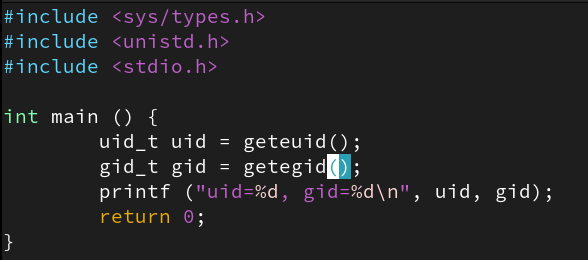


Figure 1: Исходный код файла simpleid.c

1. Скомпилируем, запустим и сравним результат с выводом команды id. User ID и Group ID совпадают.



Figure 2: Результат

1. Перепишем программу, чтобы она возвращала нам e\_uid/e\_gid и действительный uid/gid. getgid возвращает действительный идентификатор группы текущего процесса. getegid возвращает эффективный идентификатор группы текущего процесса. Действительный идентификатор соответствует идентификатору вызывающего процесса. Эффективный идентификатор соответствует биту setuid на исполняемом файле.

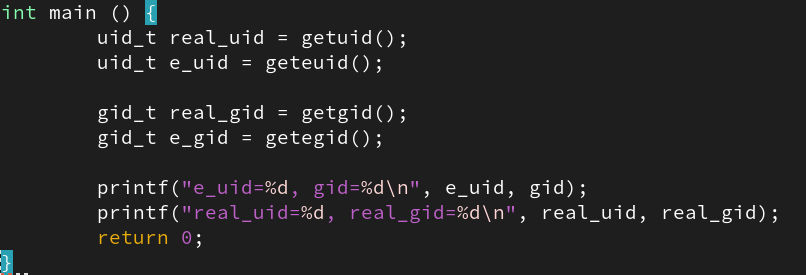


Figure 3: Измененный исходный код программы simpleid.c

1. Поменяем пользователя-обладателя исполняемого файла и добавить бит setuid.

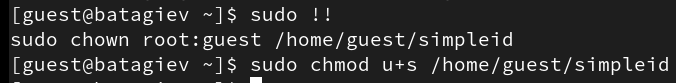


Figure 4: Изменение прав

1. Проверим изменения.

Figure 5: ls -l

Figure 5: ls -l

1. Запустим исполняемый файл.

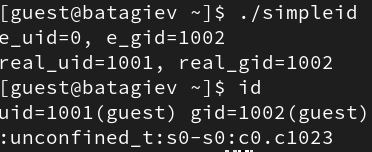


Figure 6: Результат

1. Проведем те же манипуляции над файлов, но теперь для группы.

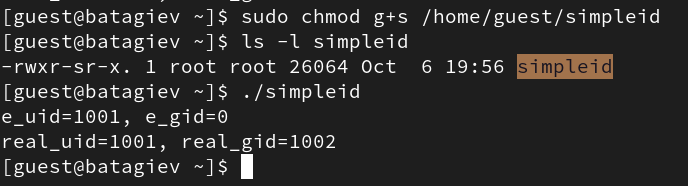


Figure 7: Результат

1. Создадим программу readfile.c

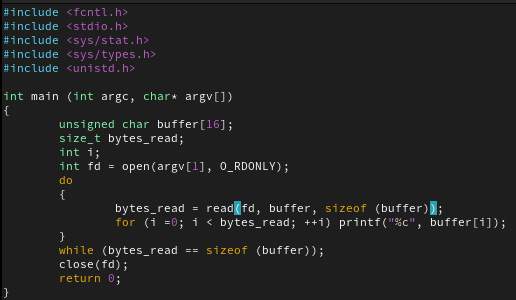


Figure 8: Исходный код файла readfile.c

1. Изменим права для файла readfile.c. Проверим изменения.

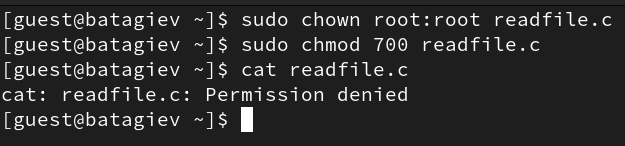


Figure 9: Измение прав файла readfile.c

1. Сменим владельца файла readfile, а также добавим setuid бит.

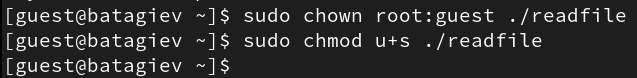


Figure 10: владелец и setuid бит

1. Проверим чтения разных файлов. Например readfile.c и /etc/shadow.

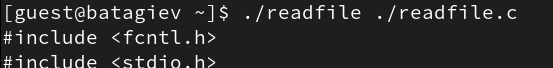


Figure 11: readfile.c

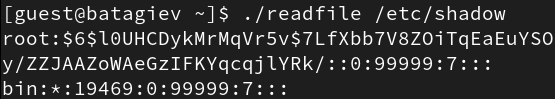


Figure 12: /etc/shadow

## 2.2 Sticky бит

1. Проверим установлен ли атрибут Sticky на директорию /tmp.

Figure 13: /tmp

Figure 13: /tmp

1. Создадим файл /tmp/file01.txt. Добавим для остальных пользователей права на чтение и запись.

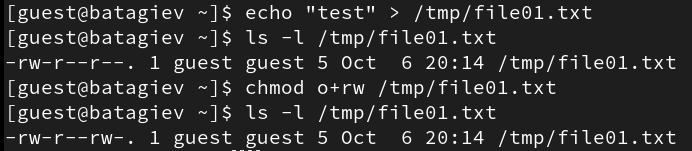


Figure 14: /tmp/file01.txt

1. Попробуем прочитать содержимое файла от пользователя guest2.

Figure 15: guest2

Figure 15: guest2

1. Допишем в конец файла новый текст.

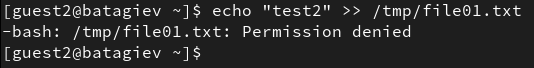


Figure 16: guest2

1. А также удалим этот файл.

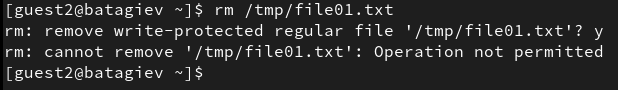


Figure 17: guest2

1. Удалим sticky бит директории /tmp.

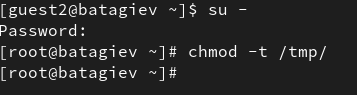


Figure 18: /tmp

1. Повторим предыдущие действия по изменению файла.

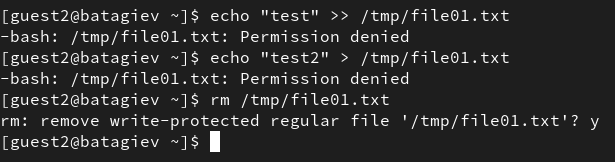


Figure 19: /tmp

# 3 Выводы

В результате выполнения работы я выполнил цели работы.