Лабораторная работа №5

Дисциплина: Основы информационной безопасности

Жибицкая Евгения Дмитриевна

Содержание

# 1 Цель работы

Продолжение работы на ОС Rocky. Изучение механизмов изменения идентификаторов, применения SetUID- и Sticky-битов и рассмотрение работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# 2 Выполнение лабораторной работы

Перед началом выполнения проверим наличие у нас gcc, а также установим setenforce на 0 (рис. 1).

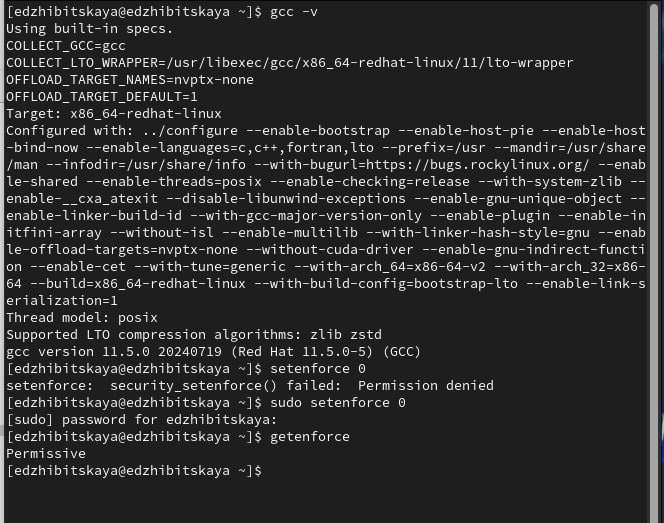


Рис. 1: Проверка gcc

Далее приступим к работе. От имени пользователя создаем файл и вставляем туда код по получению информации о пользователе(рис. 2).

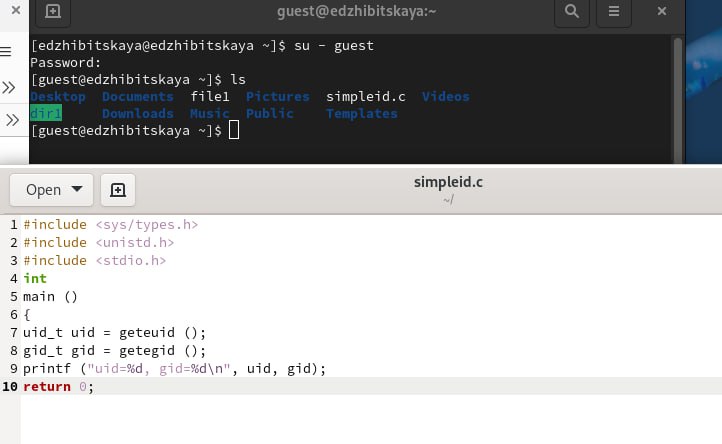


Рис. 2: Создание файла simpleid.c

Компилируем его, запускаем и смотрим на вывод. Также сравним результат программы с выводом команды id(рис. 3).



Рис. 3: Запуск файла

Доработаем код, переимнуем файл на simpleid.2 и также скомпилируем и запустим(рис. 4).

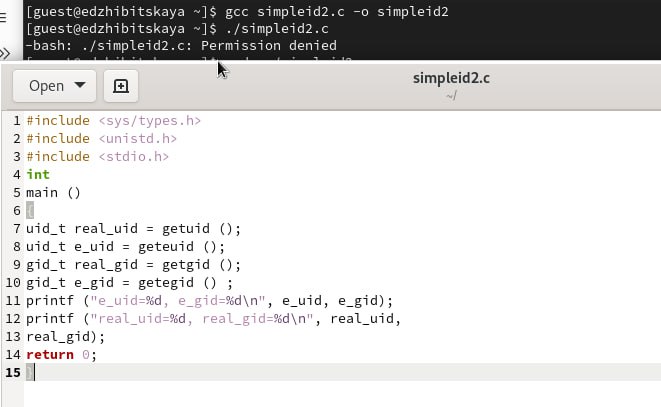


Рис. 4: Файл simpleid2.c

Далее от администратора добавим его как владельца и повысим права на этот файл(рис. 5). Проверим, что все хорошо и запустим его. Сравним с командой id(рис. 6).

Работа с правами

Рис. 5: Работа с правами

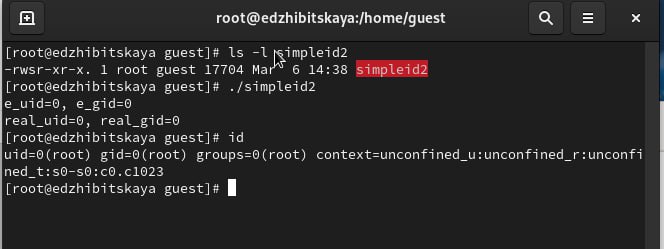


Рис. 6: Исполнение simpleid2.c

Затем создадим файл readfile, вставим код

#include <fcntl.h>  
#include <stdio.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
#include <unistd.h>  
int  
main (int argc, char\* argv[])  
{  
unsigned char buffer[16];  
size\_t bytes\_read;  
int i;  
int fd = open (argv[1], O\_RDONLY);  
do  
{  
bytes\_read = read (fd, buffer, sizeof (buffer));  
for (i =0; i < bytes\_read; ++i) printf("%c", buffer[i]);  
}  
while (bytes\_read == sizeof (buffer));  
close (fd);  
return 0;  
}

и скомпилируем его(рис. 7).

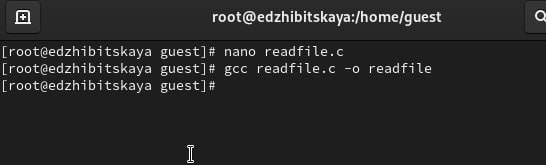


Рис. 7: Файл readfile

Сменим владельца у файла и повысим на него права(рис. 8).



Рис. 8: Работа с правами

Перейдем в пользователя guest, сменим у программы readfile владельца и установим SetU’D-бит, попробуем запустить файл - получим отказ в доступе(рис. [-fig. 9).

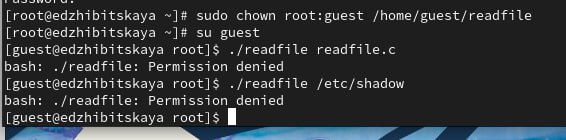


Рис. 9: Запуск файла

Перейдем к следующему заданию.

Сначала проверем установлен ли на директорию stiky-бит, запишем в него сообщение. Посмотрим на установленные права, разрешим чтение и запись для всех остальных пользователей (рис. [-fig. 10).

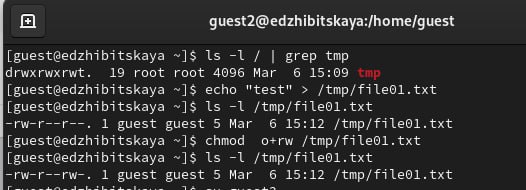


Рис. 10: Sticky-бит

От пользователя guest2 прочитаем файл, попробуем записать туда текст(безуспешно)Попробуем удалить файл - также безуспешно(рис. [-fig. 11) и рис. [-fig. 12).

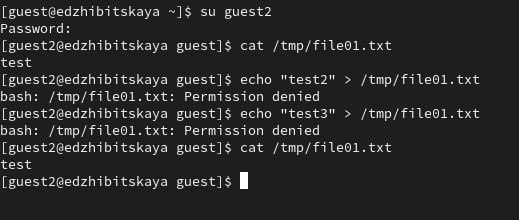


Рис. 11: Чтение от guest2

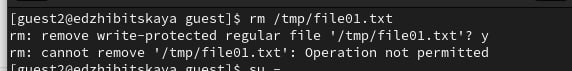


Рис. 12: Попытка удаления

Получив необходимые полномочия, уберем sticky-бит, повторим те же действия и уже удалим файл(рис. [-fig. 13).

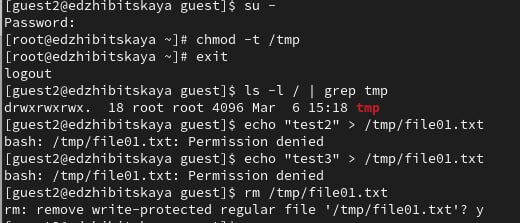


Рис. 13: Удаление файла без sticky-бита

Вернем бит и завершим выполнение работырис. [-fig. 14).

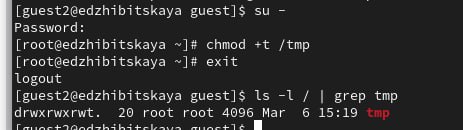


Рис. 14: Возвращение sticky-бит

# 3 Выводы

В ходе работы были изучены механизмы изменения идентификаторов, применения SetUID- и Sticky-битов а также влияние бита Sticky на запись и удаление файлов.

# Список литературы

[ТУИС](https://esystem.rudn.ru/pluginfile.php/2580984/mod_resource/content/2/005-lab_discret_sticky.pdf)