Отчёт по лабораторной работе №5

Дисциплина: Основы информационной безопасности

Паращенко Антонина Дмитриевна

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 Подготовка к выполнению лабораторной работы

1. Проверяем установлен ли компилятор gcc командой ***gcc -v*** (рис. 1)

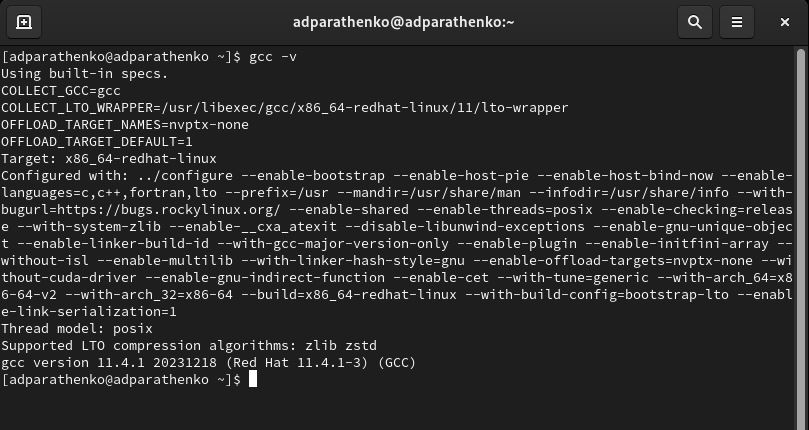


Рис. 1: gcc -v

1. Отключаем систему запретов до очередной перезагрузки системы командой ***setenforce 0*** (рис. 2)

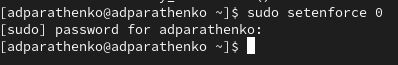


Рис. 2: setenforce

1. Проверяем наличие компиляторов С и С++ ***whereis gcc whereis g++*** (рис. 3)

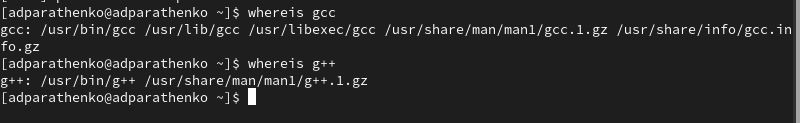


Рис. 3: компиляторы С и С++

## 2.2 Выполнение лабораторной работы

### 2.2.1 Создание программы

1. Войдите в систему от имени пользователя guest.
2. Создайте программу simpleid.c (рис. 4) - (рис. 5)

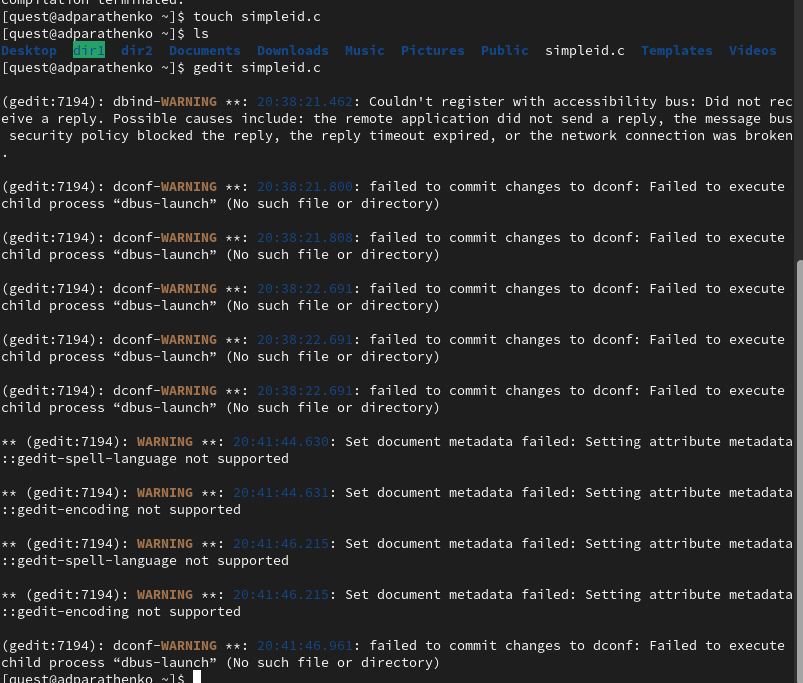


Рис. 4: Создание программы



Рис. 5: Программа simpleid.c

1. Скомплилировали программу и убедились, что файл программы создан ***gcc simpleid.c -o simpleid***
2. Выполняем программу simpleid
3. Выполняем системную программу id и сравниваем результаты.(рис. 6)

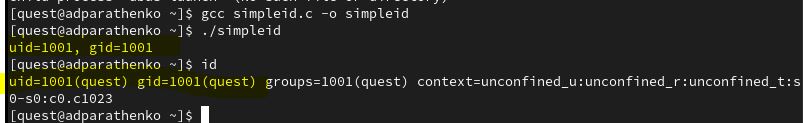


Рис. 6: Выполнение программы simpleid.c и id

1. Усложняем программу, добавив вывод действительных идентификаторов. Получившуюся программу назовите simpleid2.c. (рис. 7)

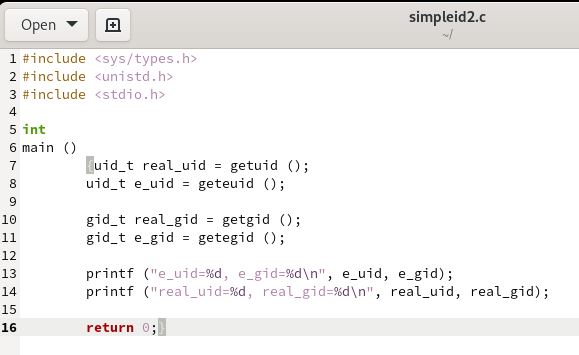


Рис. 7: Программа simpleid2.c

1. Компилируем и запускаем simpleid2.c: ***gcc simpleid2.c -o simpleid2 ./simpleid2*** (рис. 8)

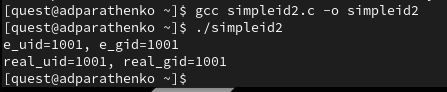


Рис. 8: Запуск программы simpleid2.c

1. От имени суперпользователя выполните команды: ***chown root:guest /home/guest/simpleid2 chmod u+s /home/guest/simpleid2***
2. Выполните проверку правильности установки новых атрибутов и смены владельца файла simpleid2: ***ls -l simpleid2*** (рис. 9)

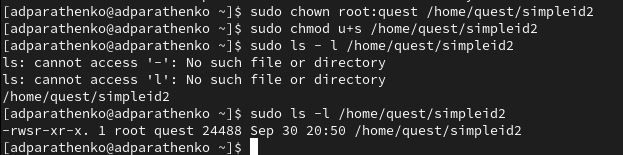


Рис. 9: Установка атрибутов

1. Запускаем simpleid2 и id: ***./simpleid2 id*** и сравниваем результаты. (рис. 10)

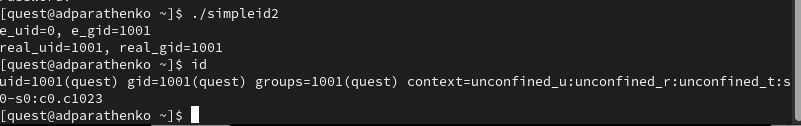


Рис. 10: Программа simpleid2.c и id

1. Создайте программу readfile.c: (рис. 11) - (рис. 12)

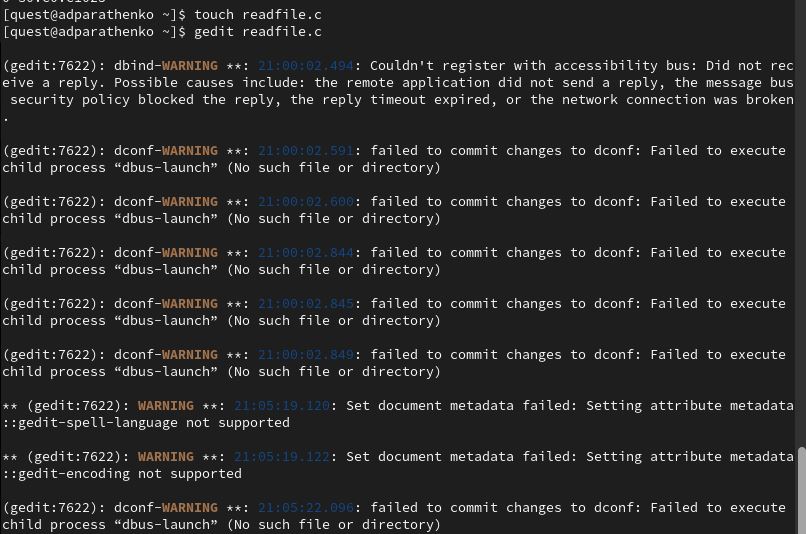


Рис. 11: Создание программы readfile.c

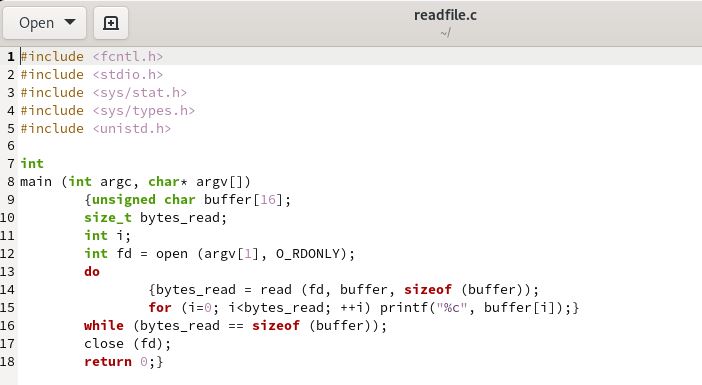


Рис. 12: Программа readfile.c

1. Откомпилируем програму readfile.c. (рис. 13)

Рис. 13: Компилирование программы readfile.c

Рис. 13: Компилирование программы readfile.c

1. Меняем владельца у файла readfile.c (или любого другого текстового файла в системе) и изменяем права так, чтобы только суперпользователь(root) мог прочитать его, a guest не мог.
2. Проверяем, что пользователь guest не может прочитать файл readfile.c (рис. 14)

Рис. 14: Чтение файла readfile.c

Рис. 14: Чтение файла readfile.c

1. Меняем у программы readfile владельца и установите SetU’D-бит.
2. Проверяем, может ли программа readfile прочитать файл readfile.c (рис. 15)



Рис. 15: Чтение файла readfile.c

1. Проверяем, может ли программа readfile прочитать файл /etc/shadow (рис. 16 - рис. 17)

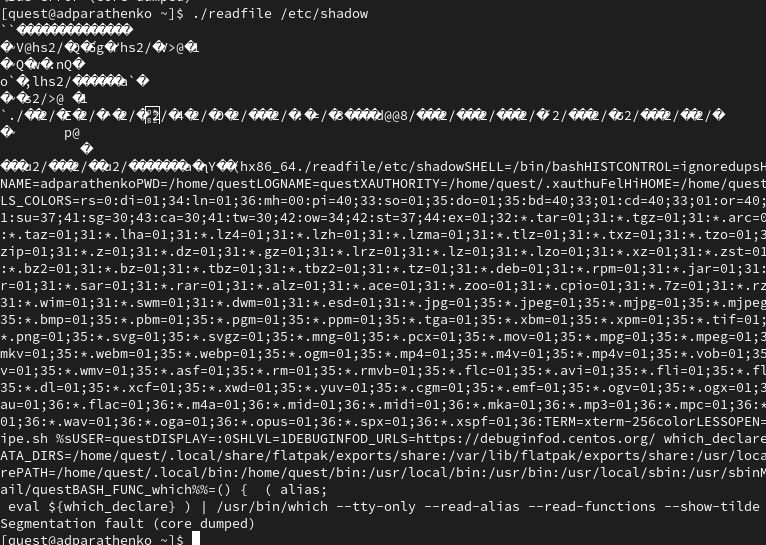


Рис. 16: Чтение файла /etc/shadow

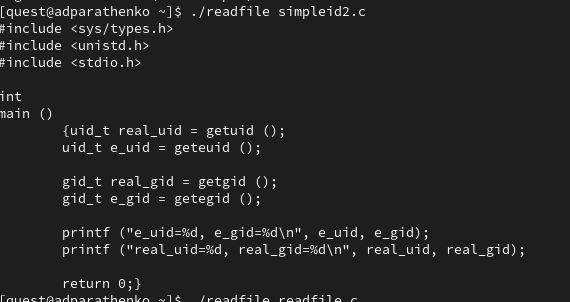


Рис. 17: Чтение файла simpleid2.c

### 2.2.2 Исследование Sticky-бита

1. Выясним, установлен ли атрибут Sticky на директории /tmp командой ***ls -l / | grep tmp***
2. От имени пользователя guest создаём файл *file01.txt* в директории /tmp со словом test: ***echo “test” > /tmp/file01.txt***
3. Просмотрим атрибуты у только что созданного файла и разрешим чтение и запись для категории пользователей «все остальные»: ***ls -l /tmp/file01.txt chmod o+rw /tmp/file01.txt ls -l /tmp/file01.txt*** (рис. 18)

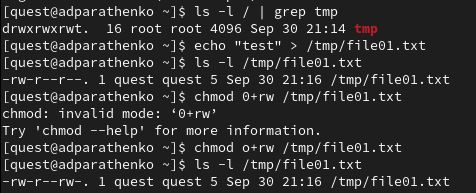


Рис. 18: Работа с файлом

4)От пользователя guest2 (не являющегося владельцем) пробуем прочитать файл */tmp/file01.txt*: ***cat /tmp/file01.txt*** 5) От пользователя guest2 пробуем дозаписать в файл */tmp/file01.txt* слово *test2* ***echo “test2” > /tmp/file01.txt*** 6) Проверяем содержимое файла ***cat /tmp/file01.txt*** 7) От пользователя guest2 пробуем записать в файл */tmp/file01.txt* слово *test3*, стерев при этом всю имеющуюся в файле информацию командой ***echo “test3” > /tmp/file01.txt*** 8) Проверяем содержимое файла командой ***cat /tmp/file01.txt*** 9) От пользователя guest2 пробуем удалить файл */tmp/file01.txt* командой ***rm /tmp/fileOl.txt*** (рис. 19)

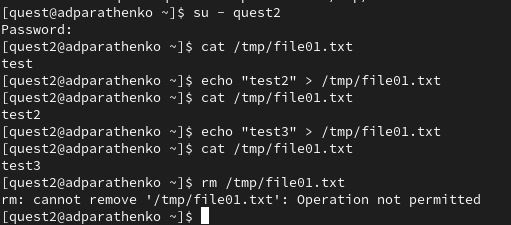


Рис. 19: Удаляем /tmp/file01.txt

1. Повышаем свои права до суперпользователя следующей командой ***su -*** и выполняем после этого команду, снимающую атрибут t (Sticky-бит) с директории /tmp: ***chmod -t /tmp***
2. Покидаем режим суперпользователя командой ***exit*** (рис. 20)

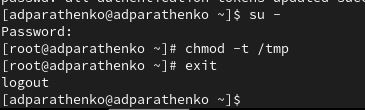


Рис. 20: Суперпользователь

1. От пользователя guest2 проверяем, что атрибута t у директории /tmp нет: ***ls -l / | grep tmp***
2. Повторяем предыдущие шаги. (рис. 21)

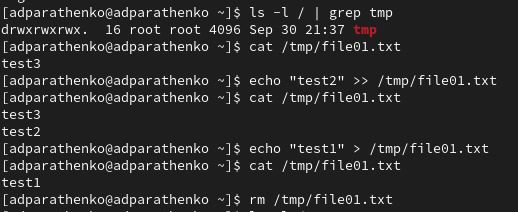


Рис. 21: Работа без атрибута t

1. Повышаем свои права до суперпользователя и возвращаем атрибут t на директорию /tmp: ***su - chmod +t /tmp exit*** (рис. 22)

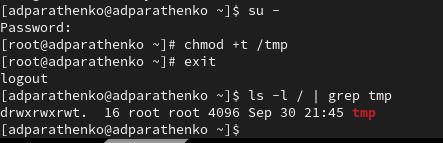


Рис. 22: Работа без атрибута t

# 3 Вывод

В результате выполнения работы я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Список литературы

1. https://esystem.rudn.ru/pluginfile.php/2357153/mod\_resource/content/2/005-lab\_discret\_sticky.pdf