Лабораторная работа №5

Дисциплина: Основы информационной безопасности

Коновалова Татьяна Борисовна

Содержание

# 1 Цель работы

Цель лабораторной работы — Изучить механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. 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.

# 2 Теоретические данные

Типы разрешений:

SetUID, SetGID и Sticku — это специальные типы разрешений, которые позволяют задавать расширенные права доступа на файлы и каталоги.

* SetUID — это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла. Другими словами, использование этого бита позволят поднять привилегии пользователя в случае, если это необходимо. Наличие SetUID бита выражается в том, что на месте классического бита x выставлен специальный бит s: -rwsr-xr-x
* SetGID — очень похож на SetUID с отличием, что файл будет запускаться от имени группы, который владеет файлом: -rwxr-sr-x
* Sticky — в случае, если этот бит установлен для папки, то файлы в этой папке могут быть удалены только их владельцем. Наличие этого бита показывается через букву t в конце всех прав: drwxrwxrwxt

Атрибуты — это набор основных девяти битов, определяющих какие из пользователей обладают правами на чтение, запись и исполнение. Первые три бита отвечают права доступа владельца, вторые — для группы пользователей, последние — для всех остальных пользователей в системе.

Установка атрибутов производится командой chmod. Установка бита чтения (r) позволяет сделать файл доступным для чтения. Наличие бита записи (w) позволяет изменять файл. Установка бита запуска (x) позволяет запускать файл на исполнение.

Расширенные атрибуты — это система дополнительной информации, которая может быть добавлена к файлу или директории в файловой системе.

Некоторые примеры расширенных атрибутов:

* а — файл можно открыть только в режиме добавления.
* А — при доступе к файлу его запись atime не изменяется.
* с — файл автоматически сжимается.
* e — файл использует экстенты.
* E — файл, каталог или символьная ссылка зашифрованы файловой системой.
* F — поиски путей в директории выполняются без учёта регистра.
* i — файл не может быть изменён.
* m — файл не сжимается.

Установка атрибутов производится командой chmod. Установка бита чтения (r) позволяет сделать файл доступным для чтения. Наличие бита записи (w) позволяет изменять файл. Установка бита запуска (x) позволяет запускать файл на исполнение.

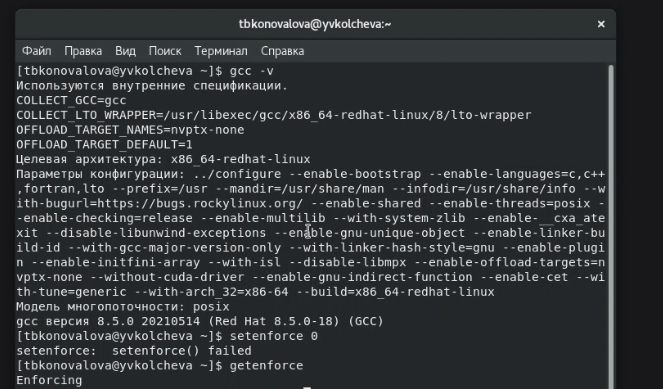
# 3 Задание

1.Создание и работа с программой simpleid.c ; 2.Исследование Sticky-бита.

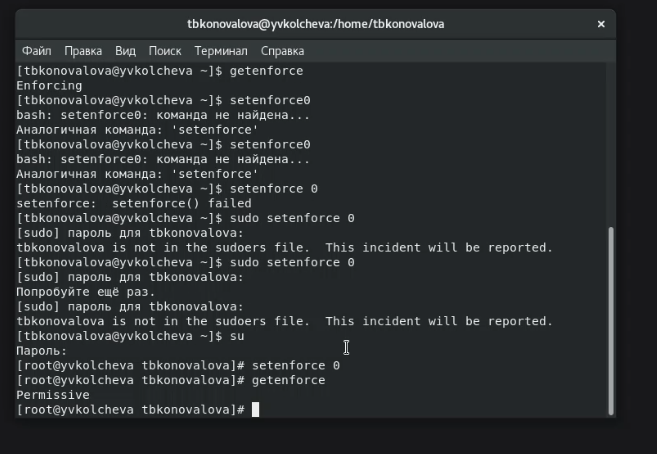
# 4 Выполнение лабораторной работы

## 4.1 Создание программы

1). Убедлась, то компилятор gcc установлен, используя команду “gcc -v”. Затем отключила систему запретов до очередной перезагрузки системы командой “sudo setenforce 0”, после чего команда “getenforce” вывела “Permissive” (рис. [??]) и (рис. [??]).

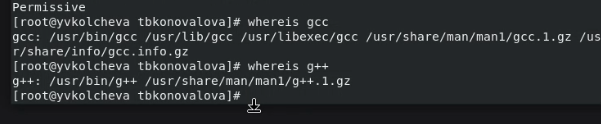


Проверка компилятора gcc



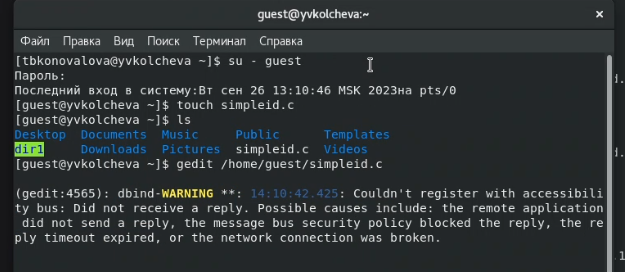
Вывод программой “Permissive”

Проверила успешное выполнение команд “whereis gcc” и “whereis g++” (их расположение) (рис. [??]).



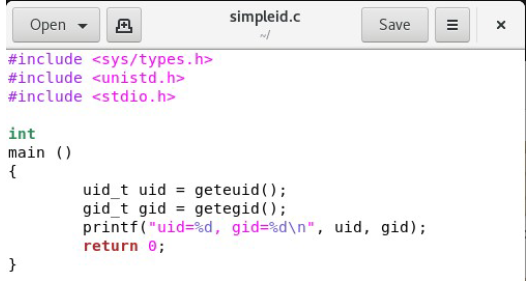
Проверка команд команд “whereis gcc” и “whereis g++”

2). Вошла в систему от имени пользователя guest командой “su - guest”. Создала программу simpleid.c командой “touch simpleid.c” и открыла её в редакторе командой “gedit /home/guest/simpleid.c” (рис. [??]).



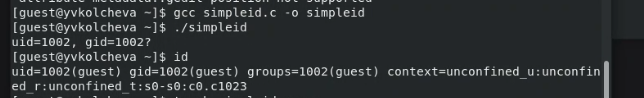
Создание программы simpleid.c

Создала программу simpleid.c со следующим текстом (рис. [??]).



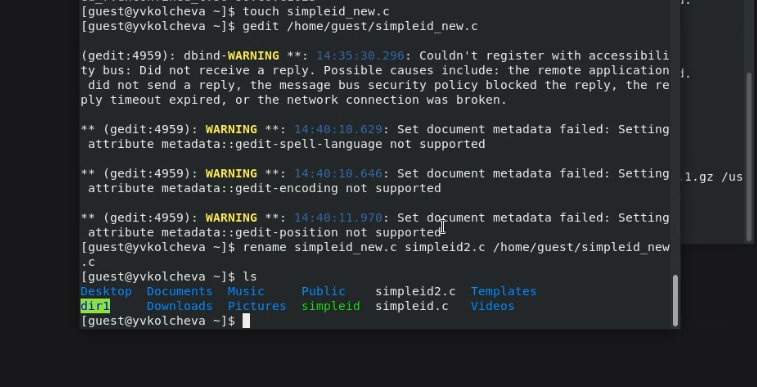
Текст программы simpleid.c

3). Скомпилировала программу с помощью команды gcc и убедилась, что файл действительно создан. Далее запустила исполняемый файл через ./. Вывод написанной программы совпадает с выводом команды id (рис [??]).



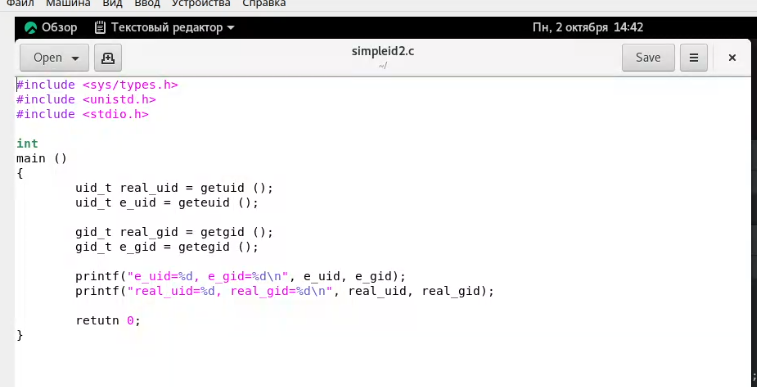
Компиляция и запуск simpleid

4). Создание файла для программы simpleid2.c и запуск данного файла для ввода программы (рис. [??]).



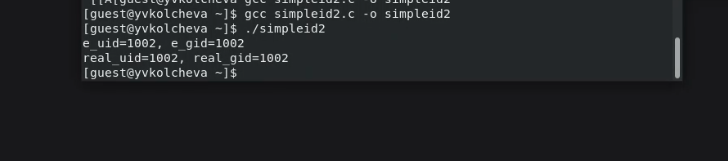
Создание файла для программы simpleid2.c

Текст усложнённой программы, назвала её simpleid2.c (рис. [??]).



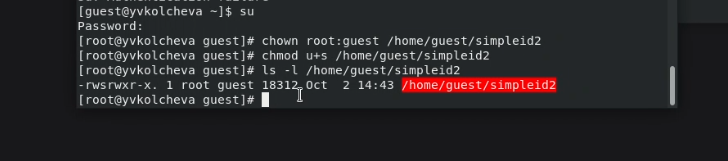
Текст программы simpleid2.c

Скомпилировала вторую программу с помощью команды gcc и убедилась, что файл действительно создан. Далее запустила исполняемый файл через ./. Вывод написанной программы совпадает с выводом команды id (рис [??]).



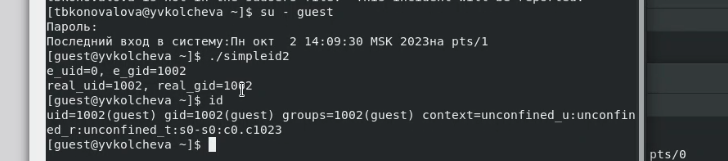
Запуск программы simpleid2.c

5). От имени суперпользователя сменила владельца файла simpleid2 на root и установила SetUID-бит. После этого через команду ls -l убедилась, что бит установился корректно (рис. [??])



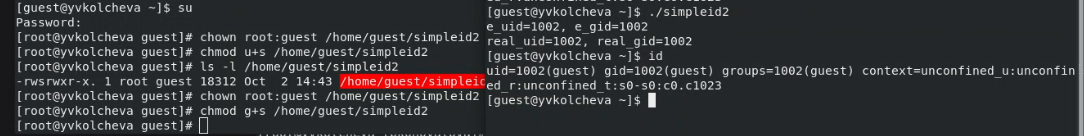
Смена владельца и установка SetUID

6). Запустила программу simpleid2 и комаду id. Теперь вижу, что появились отличия в uid строках (рис. [??]).



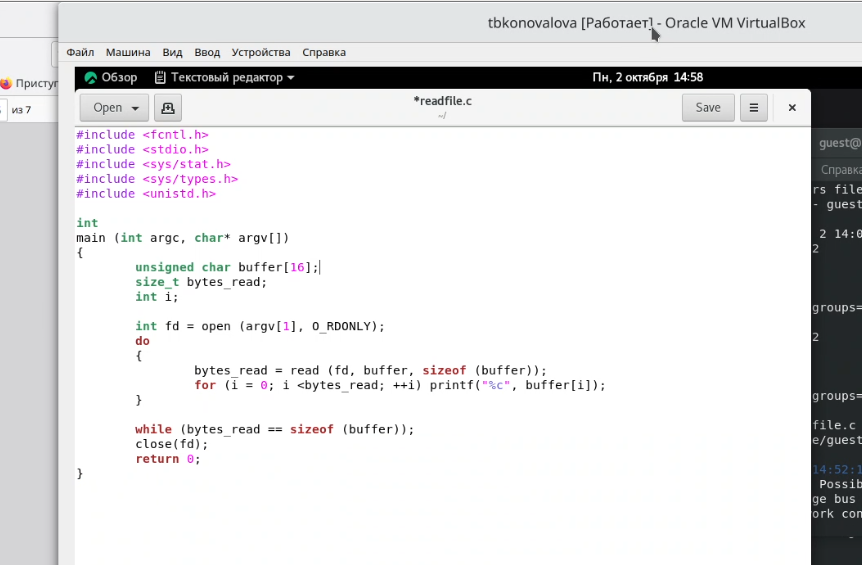
Запуск simpleid2

Проделала выше описанные действия для SetGID-бита. Теперь после запуска simpleid2 увидела отличие и в gid строках (рис. [??]).



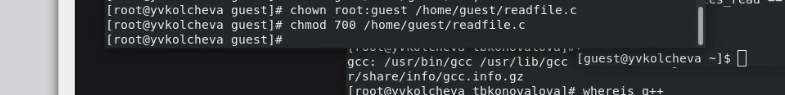
SetGID-бит

8). Создала программу readfile.c (рис. [??]).

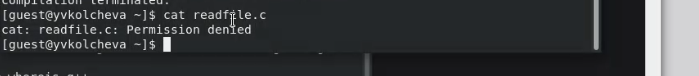


Текст программы readfile.c

Откомпилировала эту программу командой gcc . После этого изменила владельца файла readfile.c и убрала у пользователя guest право на чтение. При попытке прочитать файл от имени пользователя guest теперь возникает ошибка (рис. [??]) и (рис. [??]).

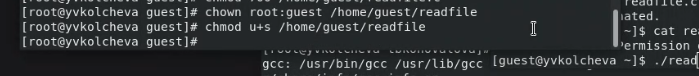


Смена владельца и прав доступа у файла readfile.c

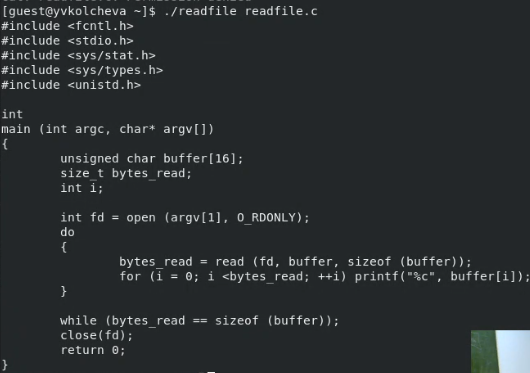


Ошибка при прочтении файла readfile.c

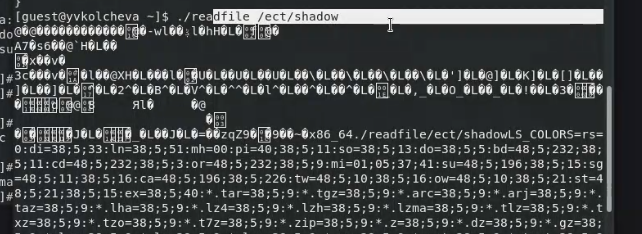
10). Поменяла владельца файла readfile и установила на него SetUID-бит (рис. [??]). Запустила исполняемый файл и убедилась, что программа может прочитать файлы readfile.c и /etc/shadow (рис. [??]) и (рис. [??]).



Меняем владельца файла readfile



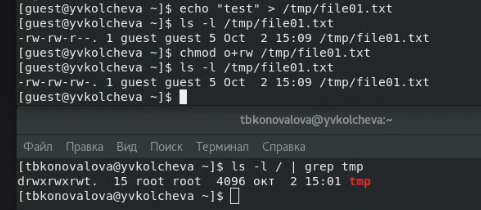
Убедиласть, что мы можем читать файл readfile



Чтение файла /etc/shadow

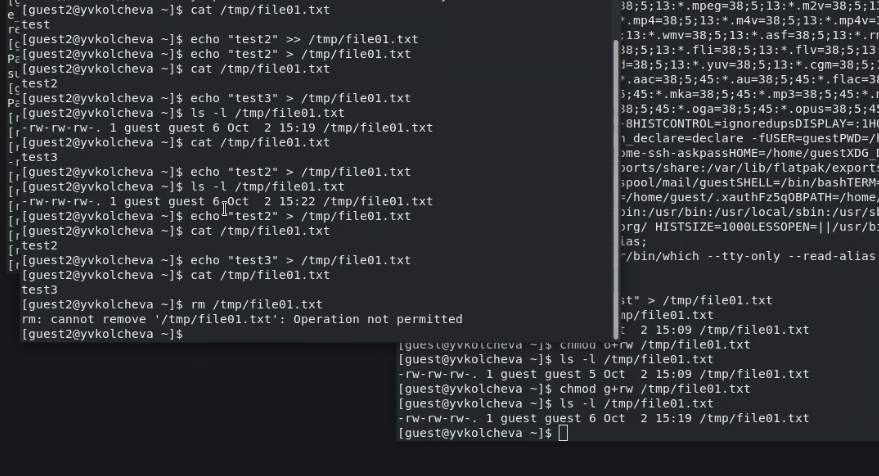
## 4.2 Исследование Sticky-бита

1). Выполняя команду ls -l выяснила, что на каталоге /tmp установлен Sticky-бит. Это видно, т.к. в конце написана t. Далее от имени пользователя guest создала файл /tmp/file01.txt. После этого просматрела атрибуты только что созданного файла и разрешила всем пользователям право на чтение и запись (рис. [??]).



Создание файла file01.txt

2). От имени пользователя guest2 прочитала файл file01.txt командой cat. Далее успешно дозаписала в конец файла строку “test2”, а затем успешно перезаписала содержимое, меняя его на строку “test3”. Однако при попытке удалить файл возникла ошибка (рис. [??]).



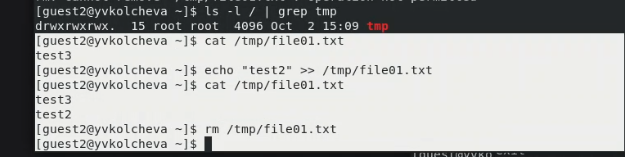
Действия над file01.txt от лица guest2

3). Временно повысела права до суперпользователя и сняла с директории /tmp Sticky-бит. Вышла из режима суперпользователя командой exit (рис. [??]).



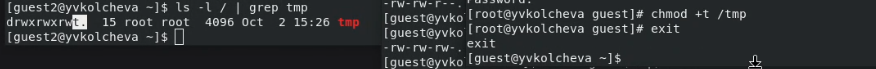
Удаление Sticky-бита

3). Убедилась с помощью команды ls -l, что Sticky-бит действительно отсутсвует. После этого повторила действия от имени пользователя guest2, описанные выше. В этот раз мне удалось удалить файл file01.txt даже при условии, что guest2 не является его владельцем (рис. [??]).



Повтор действий

4). Временно повысела права до суперпользователя и вернула Sticky-бит на каталог /tmp (рис. [??]).



Возращение Skicky-бита

# 5 Выводы

Изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. 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.

# 6 Библиография

СПИСОК ЛИТЕРАТУРЫ

1.Медведовский И.Д., Семьянов П.В., Платонов В.В. Атака через Internet. — НПО “Мир и семья-95”, 1997. — URL: http://bugtraq.ru/library/books/attack1/index.html

2.Теоеретические знания, приведённые в Лабораторной работе №5 - https://esystem.rudn.ru/pluginfile.php/2090129/mod\_resource/content/2/005-lab\_discret\_sticky.pdf

3.Запечников С. В. и др. Информационн~пасность открытых систем. Том 1. — М.: Горячаая линия -Телеком, 2006.

СПИСОК ИНТЕРНЕТ-ИСТОЧНИКОВ

1.[Электронный ресурс] - доступ: https://codeby.school/blog/informacionnaya-bezopasnost/razgranichenie-dostupa-v-linux-znakomstvo-s-astra-linux

2.[Электронный ресурс] - доступ: https://debianinstall.ru/diskretsionnoe-razgranichenie-dostupa-linux/