Лабораторная работа №5

Саргсян Арам Грачьяевич

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 Теоретическое введение

Есть 3 вида разрешений. Они определяют права пользователя на 3 действия: чтение, запись и выполнение. В Linux эти действия обозначаются вот так:

* **r** — read (чтение) — право просматривать содержимое файла;
* **w** — write (запись) — право изменять содержимое файла;
* **x** — execute (выполнение) — право запускать файл, если это программа или скрипт.

У каждого файла есть 3 группы пользователей, для которых можно устанавливать права доступа.

* **owner** (владелец) — отдельный человек, который владеет файлом. Обычно это тот, кто создал файл, но владельцем можно сделать и кого-то другого.
* **group** (группа) — пользователи с общими заданными правами.
* **others** (другие) — все остальные пользователи, не относящиеся к группе и не являющиеся владельцами[1].

Чтобы увидеть текущие назначения владельца, нужно использовать команду ls -l. Эта команда показывает пользователя и группу-владельца.

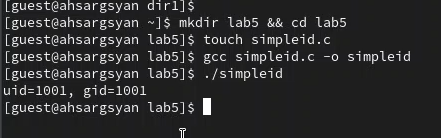
Чтобы применить соответствующие разрешения, первое, что нужно учитывать, это владение. Для этого есть команда chown[2].

Для того, чтобы позволить обычным пользователям выполнять программы от имени суперпользователя без знания его пароля была придумана такая вещь, как SUID и SGID биты. Рассмотрим эти полномочия подробнее.

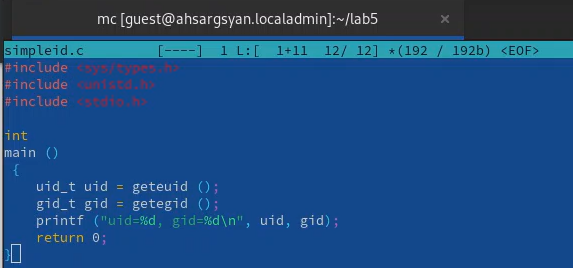
* **SUID** - если этот бит установлен, то при выполнении программы, id пользователя, от которого она запущена заменяется на id владельца файла. Фактически, это позволяет обычным пользователям запускать программы от имени суперпользователя;
* **SGID** - этот флаг работает аналогичным образом, только разница в том, что пользователь считается членом группы, с которой связан файл, а не групп, к которым он действительно принадлежит. Если SGID флаг установлен на каталог, все файлы, созданные в нем, будут связаны с группой каталога, а не пользователя. Такое поведение используется для организации общих папок;
* **Sticky-bit** - этот бит тоже используется для создания общих папок. Если он установлен, то пользователи могут только создавать, читать и выполнять файлы, но не могут удалять файлы, принадлежащие другим пользователям[3].

# 3 Выполнение лабораторной работы

1. От имени пользователя guest создал файл simpleid (рис. ??, ??).

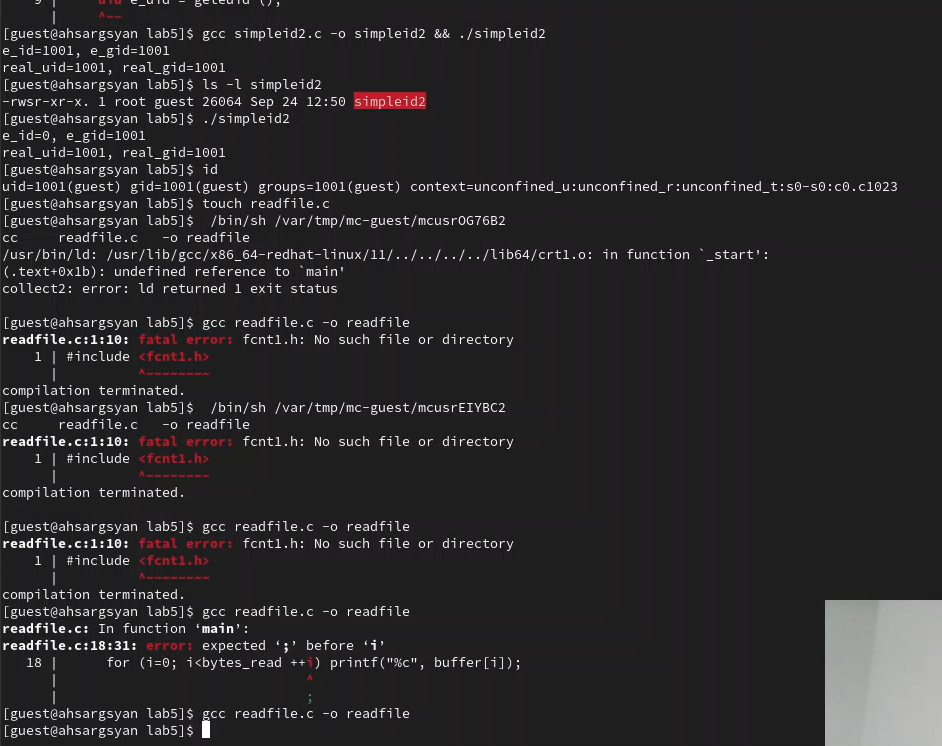


Создание файла

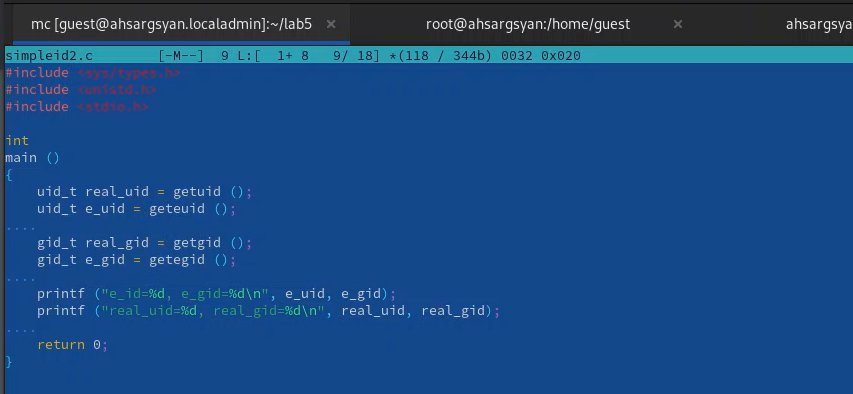


Редакция файла

1. Cоздал файл simpleid2 (рис. ??, ??).

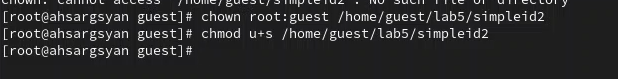


Создание файла



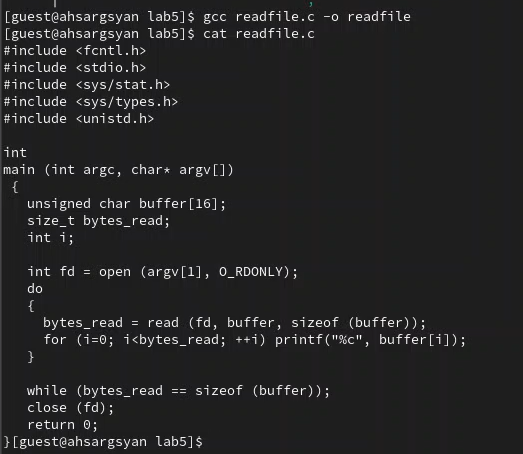
Редакция файла

1. От имени суперпользователя выполнил нужные команды (рис. ??).

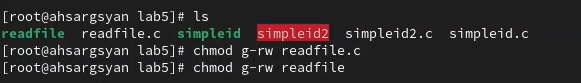


Редакция файла

1. Создал readfile.c, изменил права доступа (рис. ??, ??).

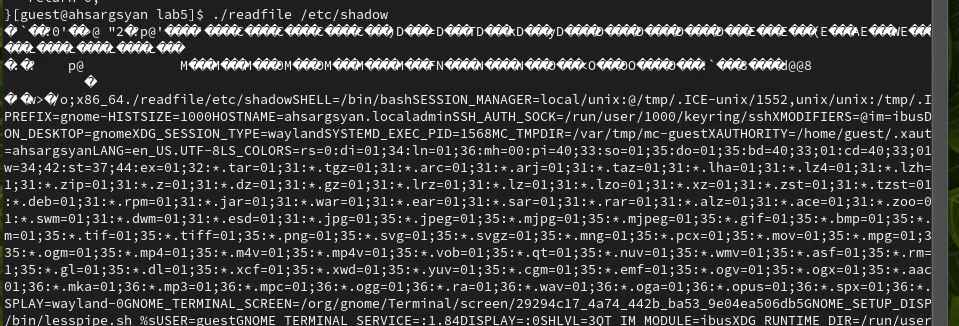


Файл readfile



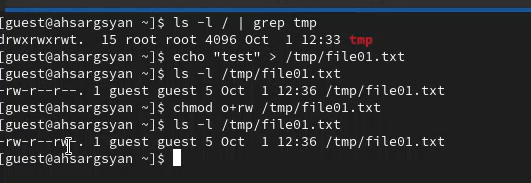
Изменение прав доступа

1. Прочитал с помощью readfile etc/shadow (рис. ??).



Изменение прав доступа

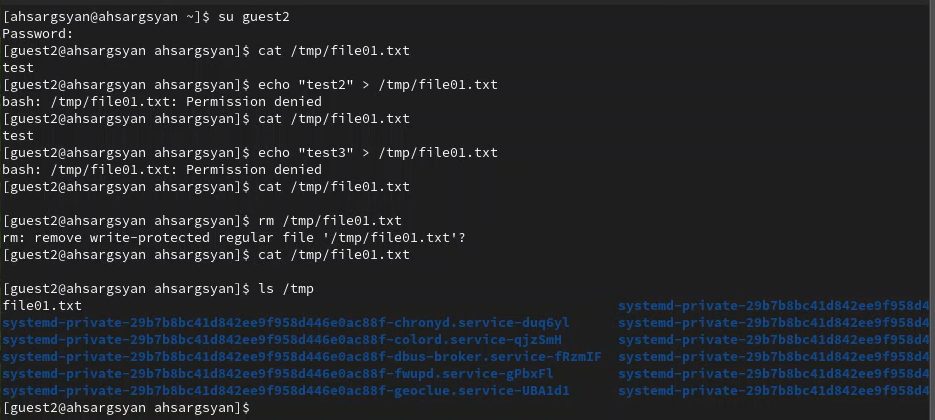
1. Исследовал sticky bit (рис. ??, ??, ??).



Пользователь guest



Пользователь guest2



Пользователь root

# 4 Выводы

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

# Список литературы

1. Права доступа в Linux [Электронный ресурс]. 2023. URL: <https://codechick.io/tutorials/unix-linux/unix-linux-permissions>.

2. Права в Linux (chown, chmod, SUID, GUID, sticky bit, ACL, umask) [Электронный ресурс]. 2023. URL: <https://habr.com/ru/articles/469667/>.

3. Права доступа к файлам в Linux [Электронный ресурс]. 2023. URL: <https://losst.pro/prava-dostupa-k-fajlam-v-linux>.