Отчёт по лабораторной работе

Лабораторная работа № 5

Акопян Изабелла Арменовна

Содержание

# 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.

# Теоретическое введение

Речь пойдет о трех битах – Setuid, Setgid и Sticky Bit. Это специальные типы разрешений позволяют задавать расширенные права доступа на файлы или каталоги.

Setuid – это бит разрешения, который позволяет пользователю запускать испол-няемый файл с правами владельца этого файла. Другими словами, использование этого бита позволяет нам поднять привилегии пользователя в случае, если этонеобходимо.

Принцип работы Setgid очень похож на setuid с отличием, что файл будет запускаться пользователем от имени группы, которая владеет файлом.

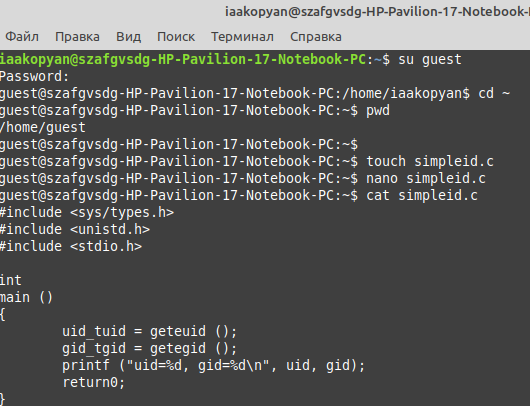
Последний специальный бит разрешения – это Sticky Bit . В случае, если этотбит установлен для папки, то файлы в этой папке могут быть удалены толькоих владельцем.

В отличие от установки sticky на каталог, на файл такой бит устанавливать уже не имеет смысла. Многие современные ядра попросту игнорируют sticky на файле. На файлах он использовался на старых системах с малой ОЗУ и был очень важен в те времена. Он запрещал выгрузку программ из памяти.

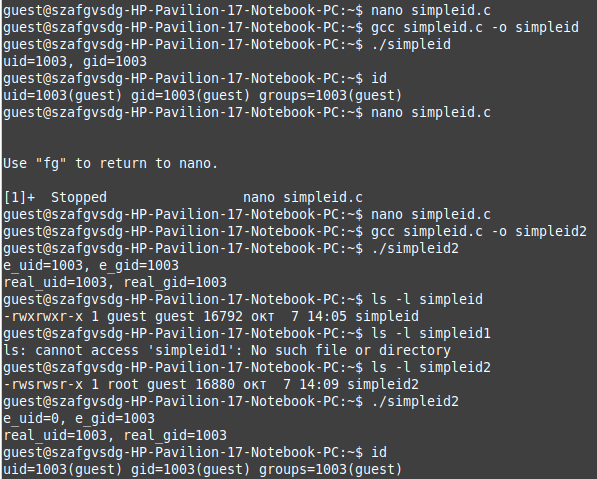
SUID работает только с файлами. Вы можете применять SGID к каталогам и файлам. Вы можете применять липкий бит только к каталогам. Если индикаторы «s», «g» или «t» отображаются в верхнем регистре, исполняемый бит (x) не установлен.

# Выполнение лабораторной работы

Создали простую программу на языке С для чтения uid и gid. Проверили ее работу, сравнив с системной утилитой id (см. рис. @fig:000@fig:001).

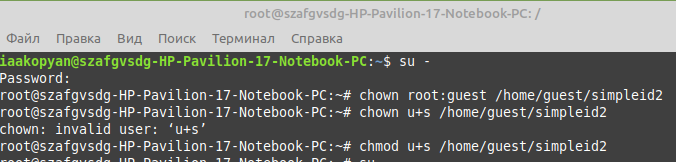


Программа для чтения uid и gid



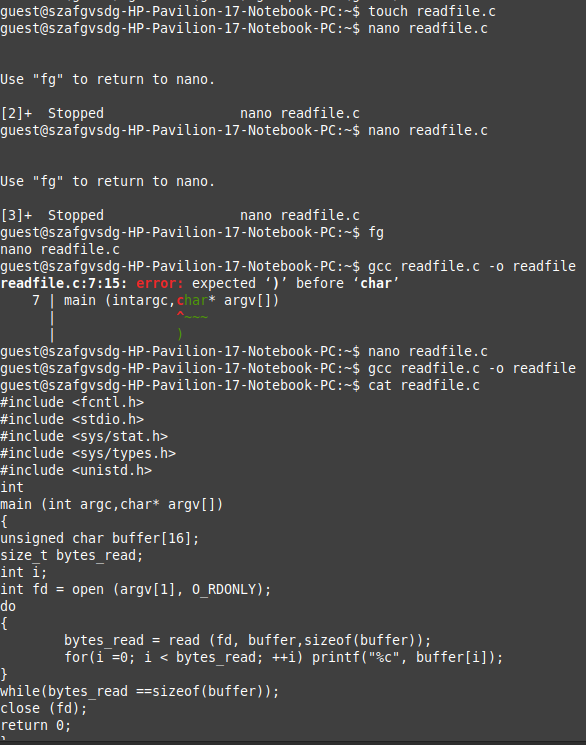
Результат работы

Усложнили программу, добавив вывод действительных идентификаторов. Командой chown root:guest /home/guest/simpleid2 передала права обладания файлов суперпользователю. Присвоила файлу SUID и GUID биты (см. рис. @fig:002). Проверили функционирование. И правда, при выполнении с правами суперпользователя uid указывается суперпользователя Так как мы не меняли группу файла, изменений в программе при выставлении GUID не наблюдается.

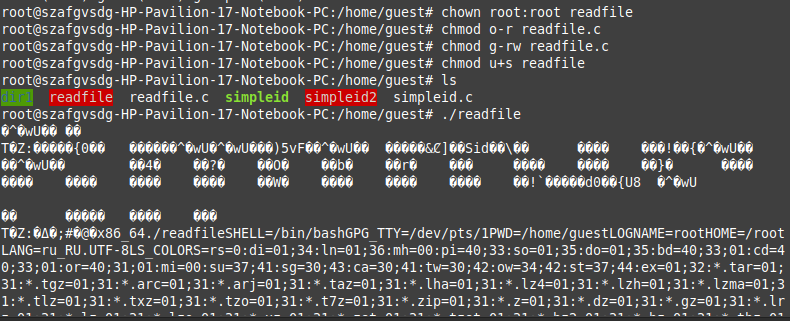


chown и chmod

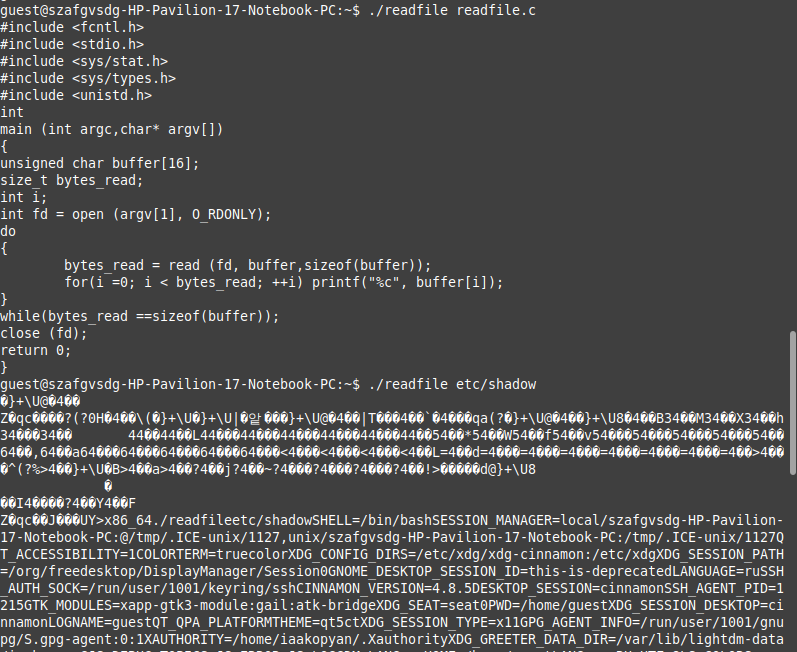
Создали программу для чтения файлов. Установили на текст программы права доступа 000. Установили владельцем исполняемого файла суперпользователя. Поставили SUID бит. Запустили исполняемый файл от обычного пользователя. Исполняемый файл смог прочесть текст программы, у которого отсутствует разрешение для чтения (см. рис. @fig:003@fig:006).



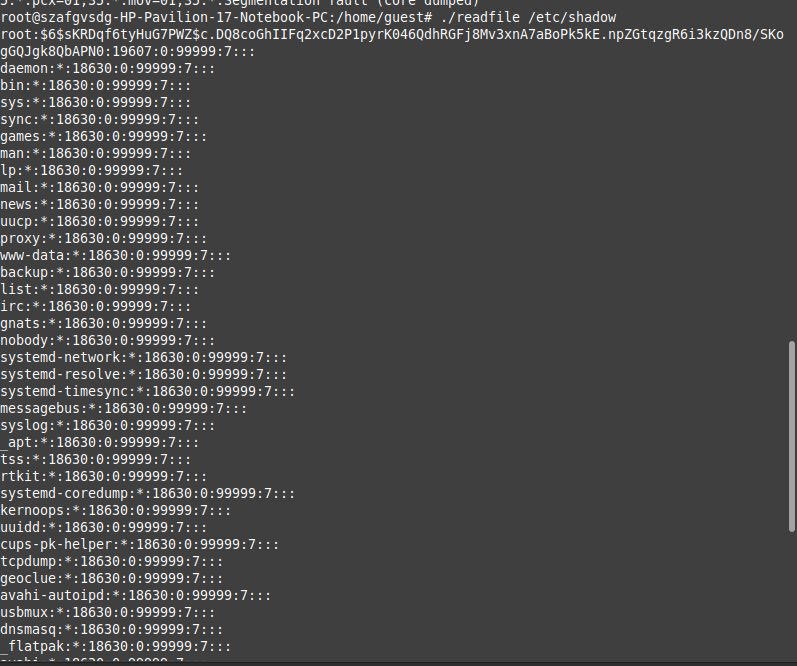
readfile.c



смена владельца



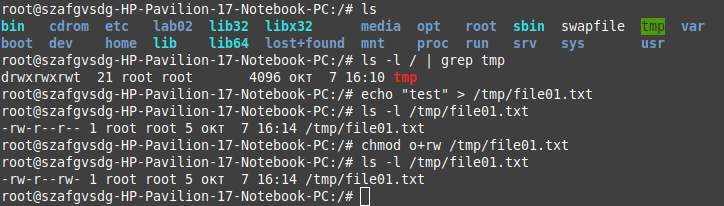
чтение readfile.c



etc/shadow

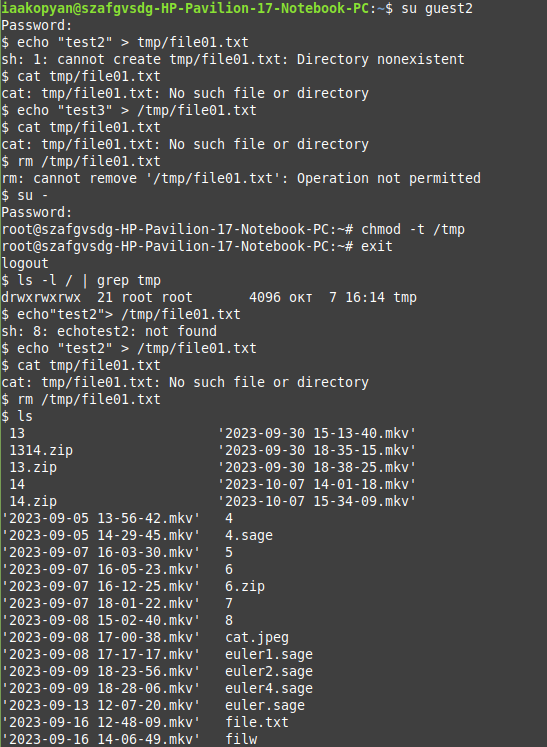
Далее поработали с Sticky-битом

Создала файл, добавила права остальным пользователям.



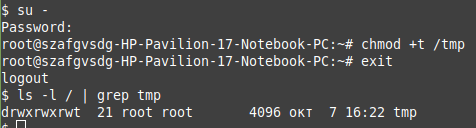
Проверка функциональности Sticky-бита на примере файла, созданного в каталоге /tmp

От второго гостя попробовала записать в файл текст, удалить файл. Потом убрала атрибут t и сделала все то же самое. (см. рис. @fig:008)



атрибут t убран

Вернула все на место (см. рис. @fig:009)



атрибут t добавлен

# Выводы

Изучили теорию механизмов изменения идентификаторов, применения SetUID, SetGID и Sticky-битов. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

# Список литературы

Лабораторная работа № 5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов. [Электронный ресурс]. <https://esystem.rudn.ru/pluginfile.php/2090417/mod_resource/content/2/005-lab_discret_sticky.pdf>

Использование SETUID, SETGID и Sticky bit для расширенной настройки прав доступа в операционных системах Linux. Исследование влияния дополнительных атрибутов. [Электронный ресурс].<https://ruvds.com/ru/helpcenter/suid-sgid-sticky-bit-linux/>