Отчёт по лабораторной работе №05

Дисциплина: Основы информационной безопасности

Бансимба Клодели Дьегра НПИбд-02-22

Содержание

# 1 5.1 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 5.2 Порядок выполнения работы

## 2.1 5.2.1 Создание программы

1. Вошли в систему от имени пользователя guest.

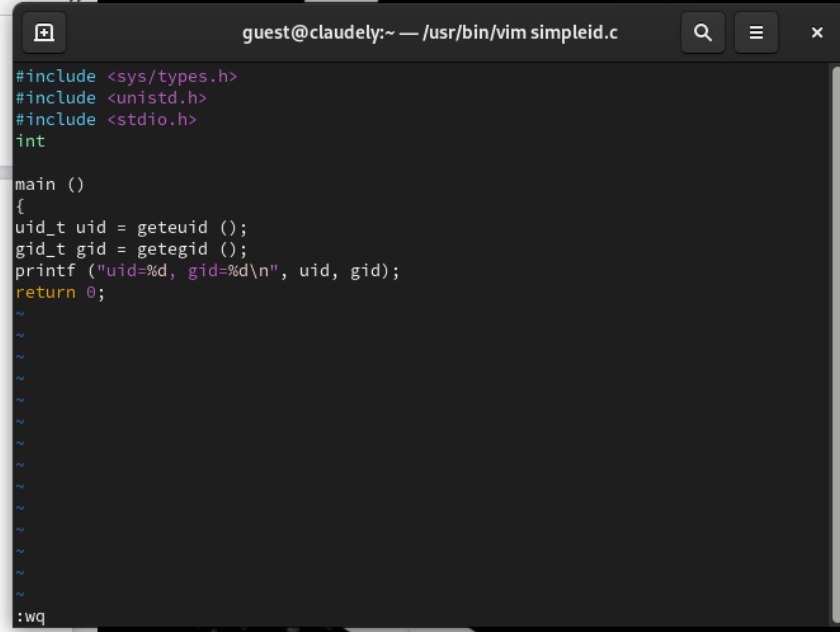


Рис. 1: программа simpleid

1. Написали программу simpleid.c.Скомпилировали программу и убедились, что файл программы создан: gcc simpleid.c -o simpleid. Выполнили программу simpleid командой ./simpleid. Выполнили системную программу id с помощью команды id. uid и gid совпадает в обеих программах

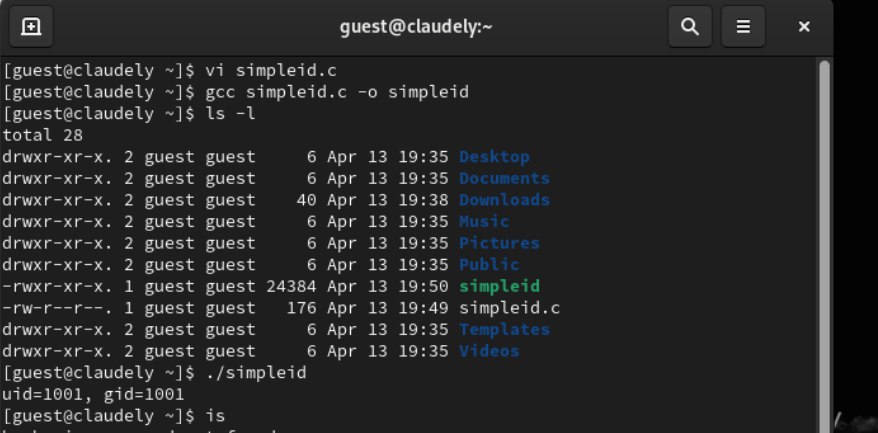


Рис. 2: программа simpleid

1. Усложнили программу, добавив вывод действительных идентификаторов.



Рис. 3: программа simpleid2

1. Скомпилировали и запустили simpleid2.c:



Рис. 4: запустили simpleid2

1. От имени суперпользователя выполнили команды, Использовали su для повышения прав до суперпользователя.Выполнили проверку правильности установки новых атрибутов и смены владельца файла simpleid2: Запустили simpleid2 и id:

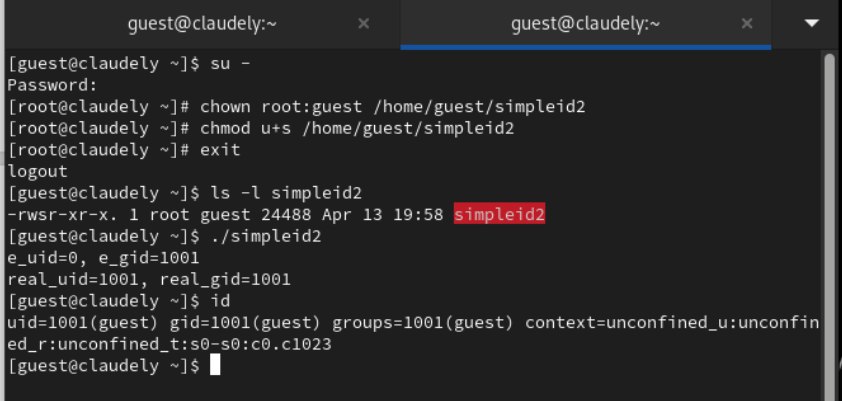


Рис. 5: запустили simpleid2 и id

1. Проделали тоже самое относительно SetGID-бита.Написали программу readfile.c

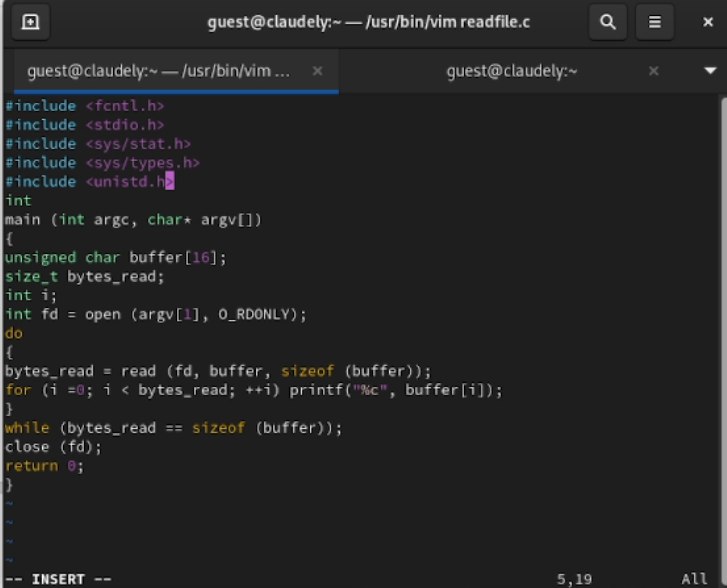


Рис. 6: программа readfile

1. Откомпилировали её.

gcc readfile.c -o readfile

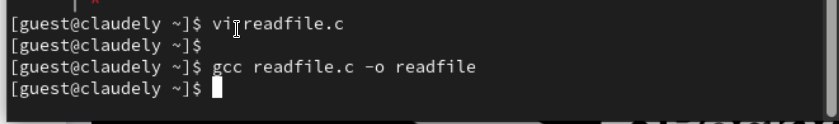


Рис. 7: программа readfile

1. Сменили владельца у файла readfile.c и изменили права так, чтобы только суперпользователь (root) мог прочитать его, a guest не мог.Проверили, что пользователь guest не может прочитать файл readfile.c .Сменили у программы readfile владельца и установили SetU’D-бит.Проверили, может ли программа readfile прочитать файл readfile.c Проверили, может ли программа readfile прочитать файл /etc/shadow

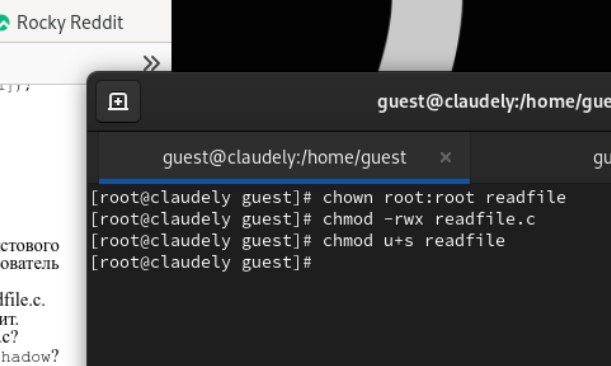


Рис. 8: результат программы readfile

##Исследование Sticky-бита 1. Выяснили, установлен ли атрибут Sticky на директории /tmp:

ls -l / | grep tmp

1. От имени пользователя guest создали файл file01.txt в директории /tmp со словом test:

echo "test" > /tmp/file01.txt

1. Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные»:

ls -l /tmp/file01.txt  
chmod o+rw /tmp/file01.txt  
ls -l /tmp/file01.txt

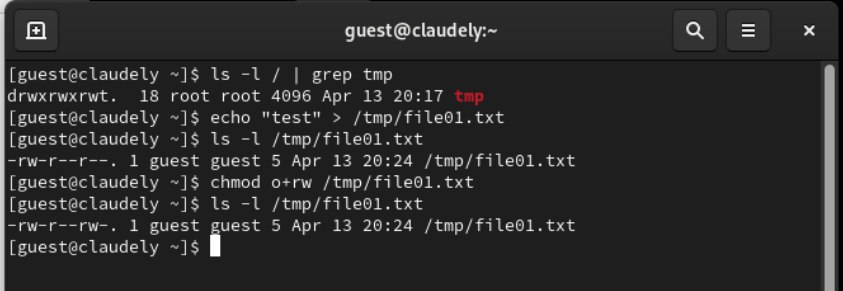


Рис. 9: исследование Sticky-бита

1. От пользователя (не являющегося владельцем) попробовали прочитать файл /file01.txt:

cat /file01.txt

1. От пользователя попробовали дозаписать в файл /file01.txt слово test3 командой:

echo "test2" >> /file01.txt

1. Проверили содержимое файла командой:

cat /file01.txt

В файле теперь записано:

Test  
Test2

1. От пользователя попробовали записать в файл /tmp/file01.txt слово test4, стерев при этом всю имеющуюся в файле информацию командой. Для этого воспользовалась командой echo “test3” > /tmp/file01.txt
2. Проверили содержимое файла командой

cat /tmp/file01.txt

1. От пользователя попробовали удалить файл /tmp/file01.txt командой rm /tmp/file01.txt, однако получила отказ.
2. От суперпользователя командой выполнили команду, снимающую атрибут t (Sticky-бит) с директории /tmp:

chmod -t /tmp

Покинули режим суперпользователя командой exit.

1. От пользователя проверили, что атрибута t у директории /tmp нет:

ls -l / | grep tmp

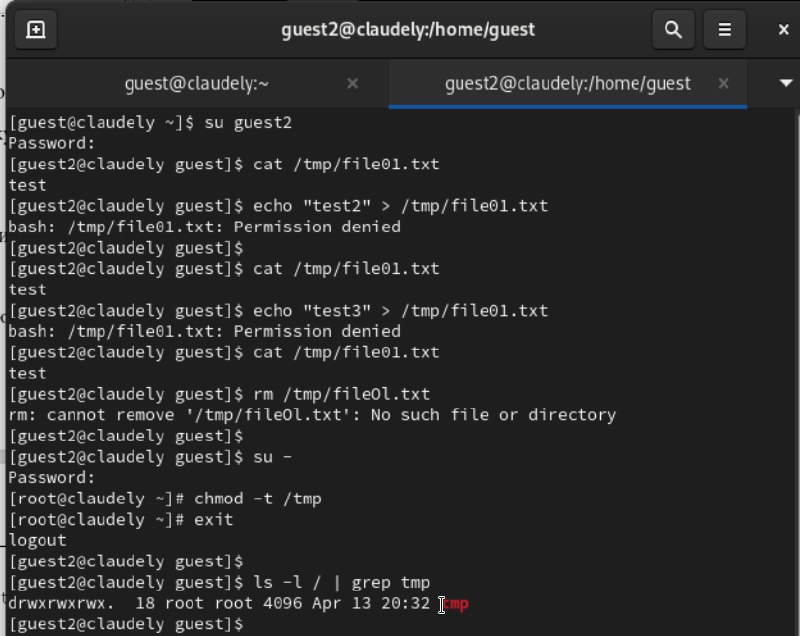


Рис. 10: исследование Sticky-бита

1. Повторили предыдущие шаги. Получилось удалить файл
2. Удалось удалить файл от имени пользователя, не являющегося его владельцем.
3. Повысили свои права до суперпользователя и вернули атрибут t на директорию /tmp :

su  
  
chmod +t /tmp  
  
exit

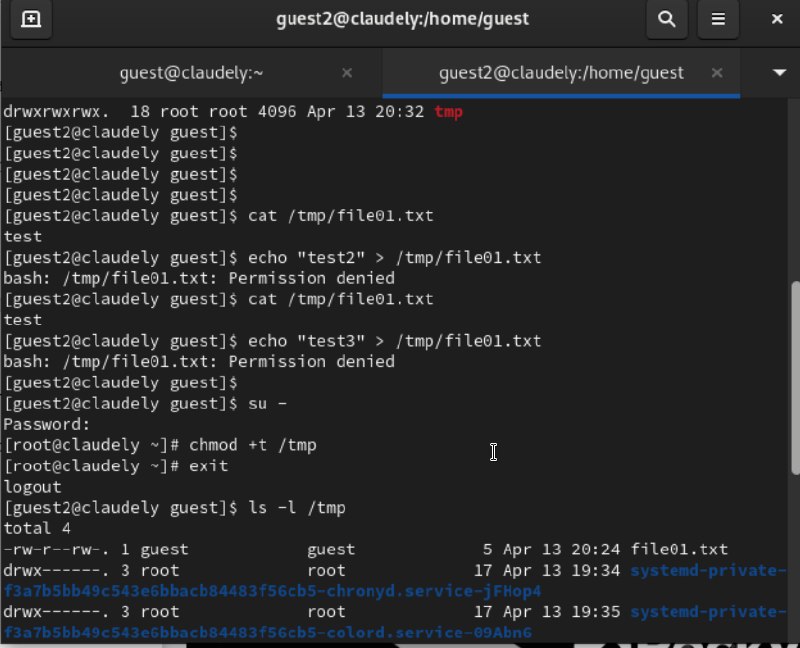


Рис. 11: исследование Sticky-бита

# 3 Выводы

Изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами.