Отчёт по лабораторной работе №6

дисциплина: Информационная безопасность

Никитаева Александра Семеновна, НПИбд-02-18

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.

# Выполнение лабораторной работы

1. Создание программы

1.1. Вошла в систему от имени пользователя guest.

1.2. Создала программу *simpleid.c* по шаблону из методички. (рис. 1)

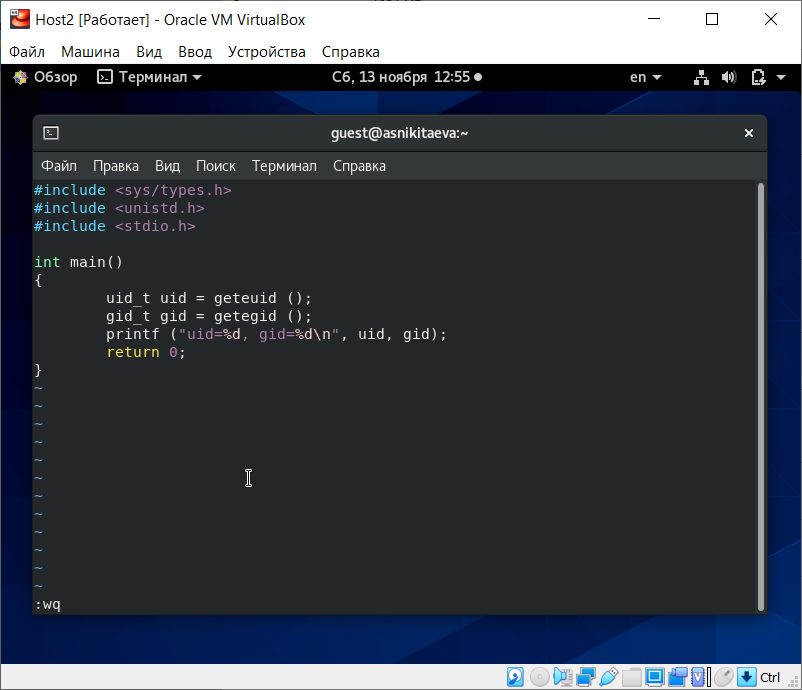


Figure 1: Программа *simpleid.c*

1.3. Скомплилировала программу и убедилась, что файл программы создан: gcc simpleid.c -o simpleid. (рис. 2)

1.4. Выполнила программу simpleid: ./simpleid. (рис. 2)

1.5. Выполнила системную программу id: id. (рис. 2) Полученный мной результат совпадает с данными предыдущего пункта задания.

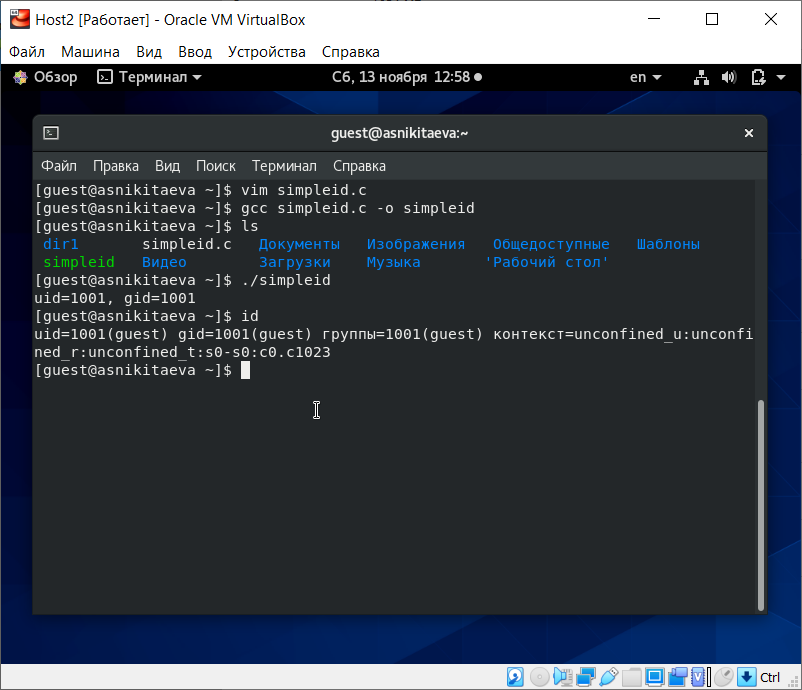


Figure 2: Компиляция и выполнение программы simpleid

1.6. Усложнила программу, добавив вывод действительных идентификаторов согласно шаблону из методички. Получившуюся программу назвала simpleid2.c. (рис. 3)

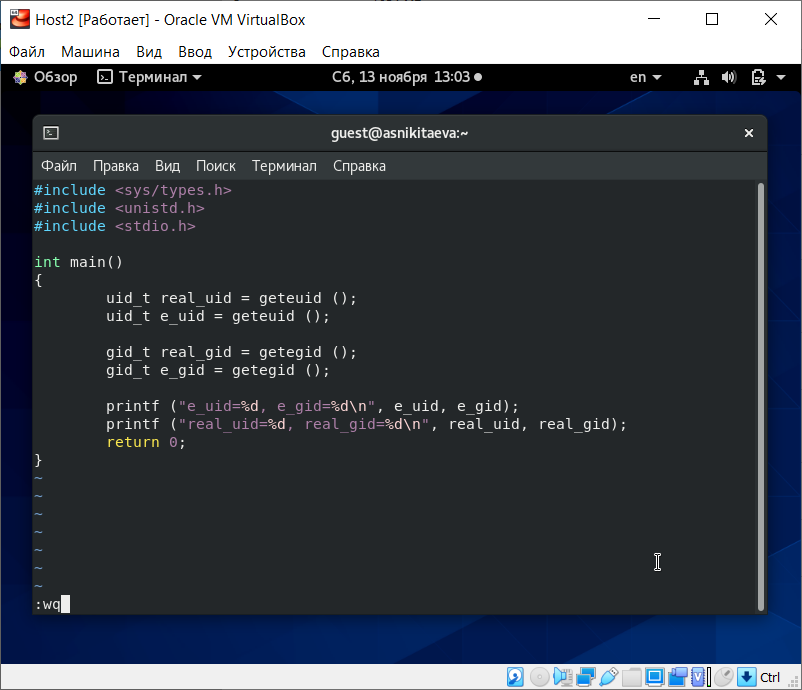


Figure 3: Программа *simpleid2.c*

1.7. Скомпилировала и запустила simpleid2.c: gcc simpleid2.c -o simpleid2 и ./simpleid2. (рис. 4)

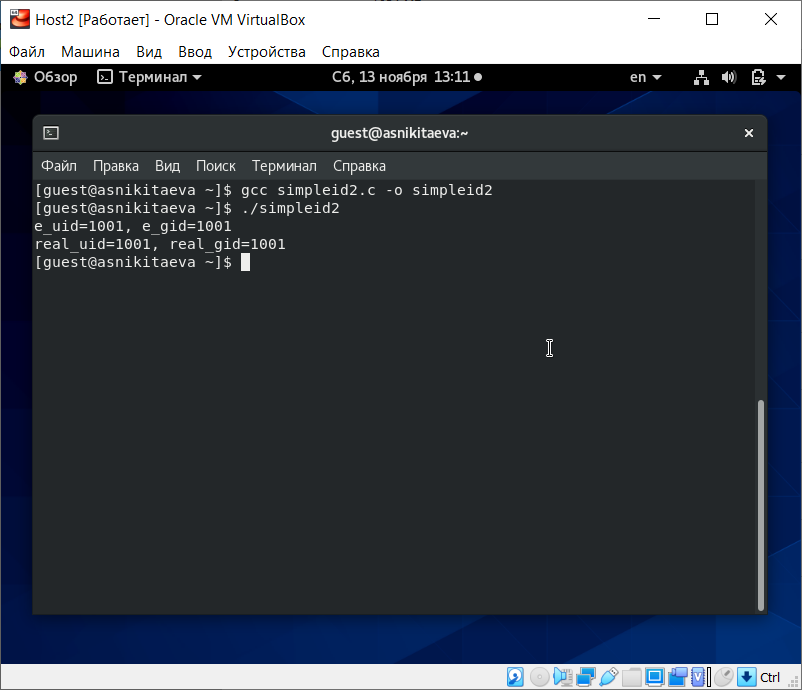


Figure 4: Компиляция и выполнение программы simpleid2

1.8. От имени суперпользователя выполнила команды: chown root:guest /home/guest/simpleid2 и chmod u+s /home/guest/simpleid2. (рис. 5)

1.9. Повысила временно свои права с помощью su. (рис. 5) Первая команда меняет владельца файла, а вторая добавляет SetUID-бит.

1.10. Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2: ls -l simpleid2. (рис. 5)

1.11. Запустила simpleid2 и id: ./simpleid2 и id. (рис. 5)

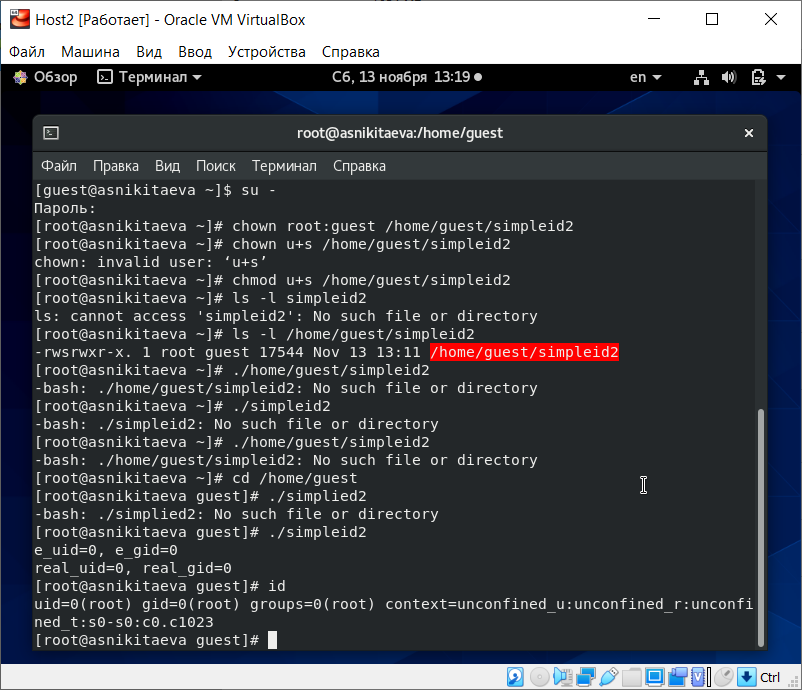


Figure 5: Смена пользователя. Установка SetUID-бита. Выполнение программы simpleid2

1.12. Проделала то же самое относительно SetGID-бита. (рис. 6)

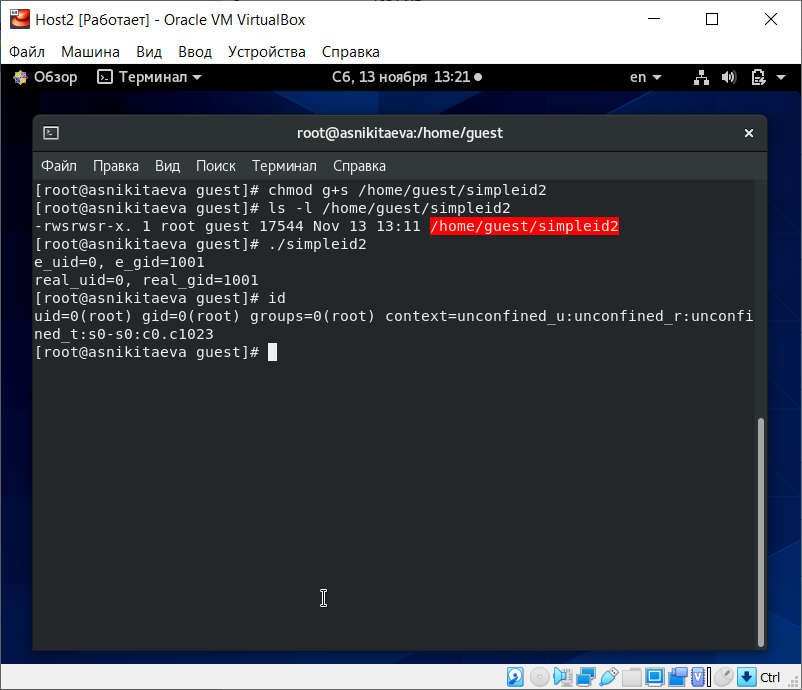


Figure 6: Установка SetGID-бита. Выполнение программы simpleid2

1.13. Создала программу readfile.c по шаблону из методички. (рис. 7)

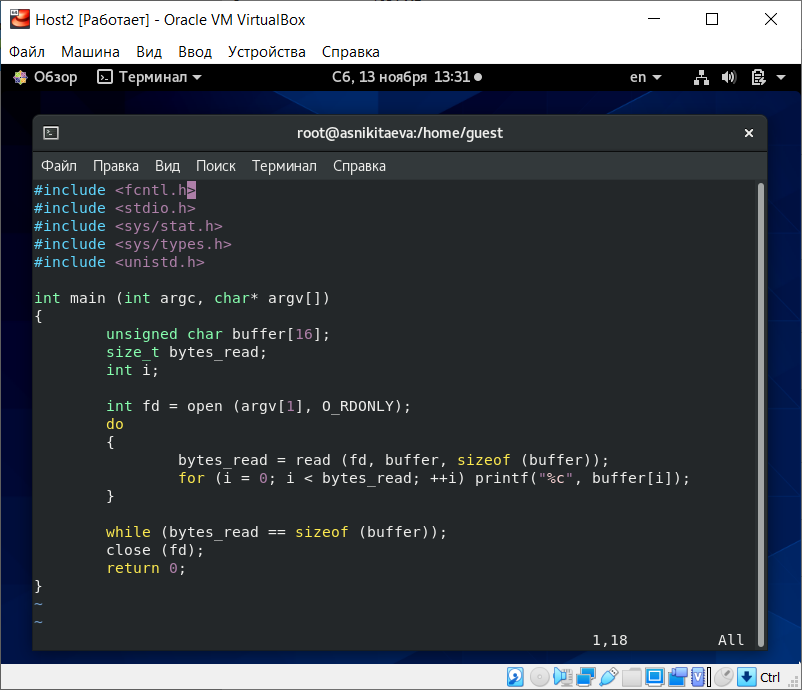


Figure 7: Программа *readfile.c*

1.14. Откомпилировала её: gcc readfile.c -o readfile. (рис. 8)

1.15. Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь (root) мог прочитать его, a guest не мог. (рис. 8)

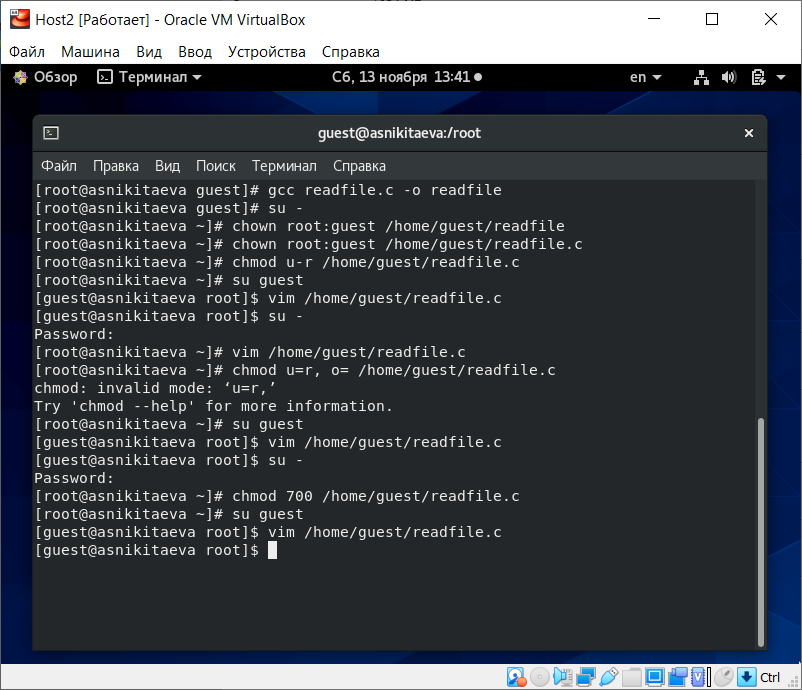


Figure 8: Работа с программой *readfile.c*

1.16. Проверила, что пользователь guest не может прочитать файл readfile.c. (рис. 9)

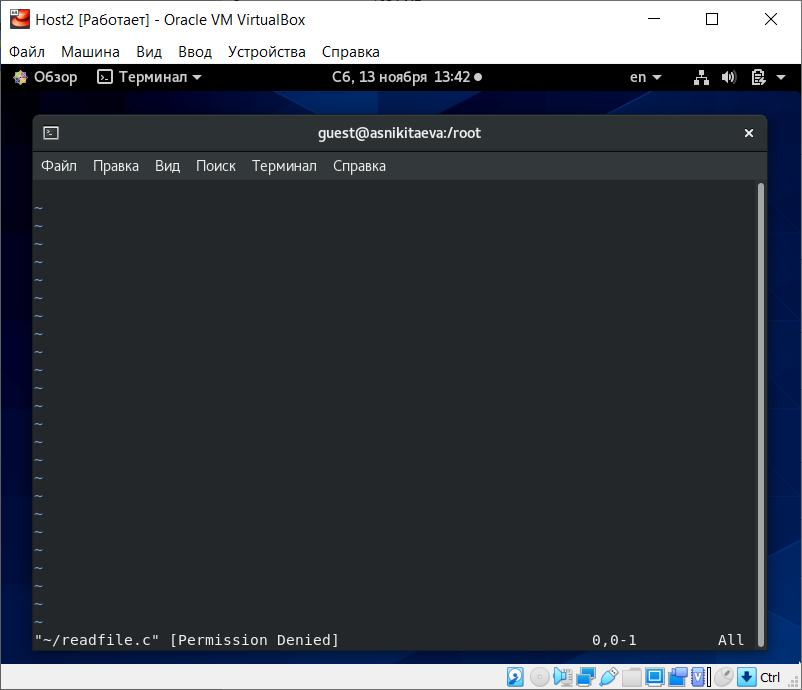


Figure 9: Запрет на чтение программы *readfile.c* для guest

1.17. Сменила у программы readfile владельца (рис. 8) и установила SetUID-бит (рис. 10).

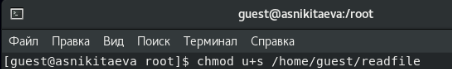


Figure 10: Установка SetUID-бита на программу readfile

1.18. Проверила, может ли программа readfile прочитать файл *readfile.c*. (рис. 11)

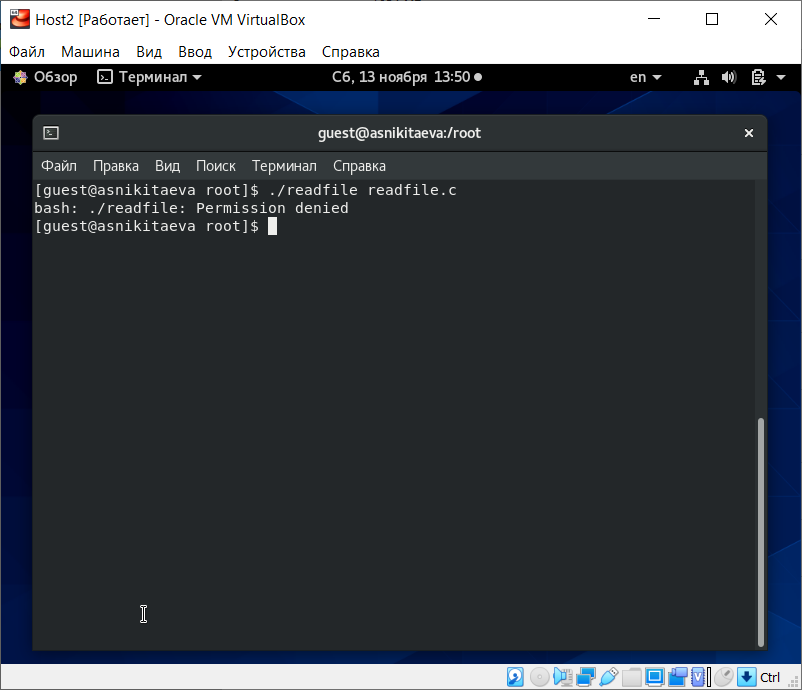


Figure 11: Программа readfile читает *readfile.c*

1.19. Проверила, может ли программа readfile прочитать файл */etc/shadow*. (рис. 12)

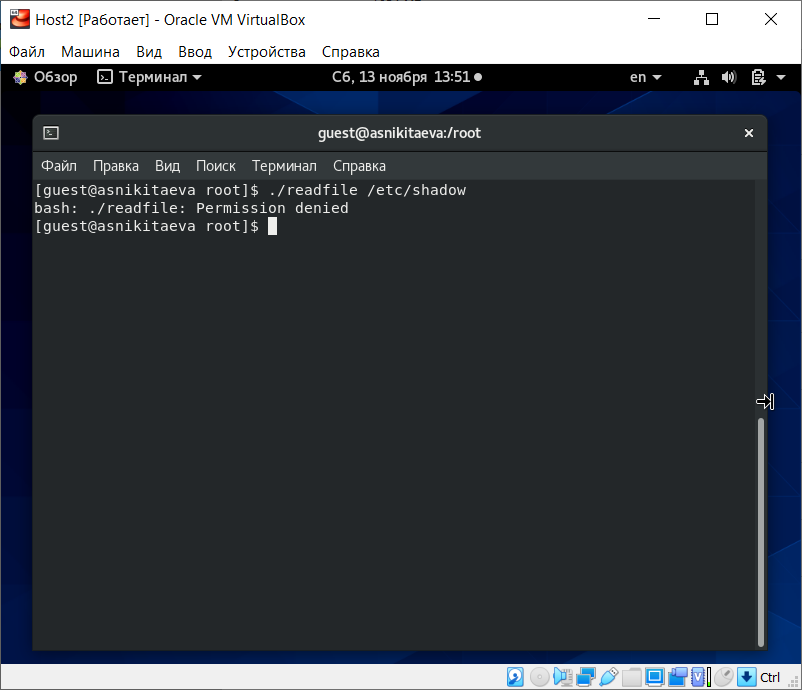


Figure 12: Программа readfile читает */etc/shadow*

1. Исследование Sticky-бита

2.1. Выяснила, установлен ли атрибут Sticky на директории /tmp, для чего выполнила команду: ls -l / | grep tmp. (рис. 13)

2.2. От имени пользователя guest создала файл file01.txt в директории /tmp со словом test: echo "test" > /tmp/file01.txt. (рис. 13)

2.3. Просмотрела атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные»: ls -l /tmp/file01.txt, chmod o+rw /tmp/file01.txt и ls -l /tmp/file01.txt. (рис. 13)

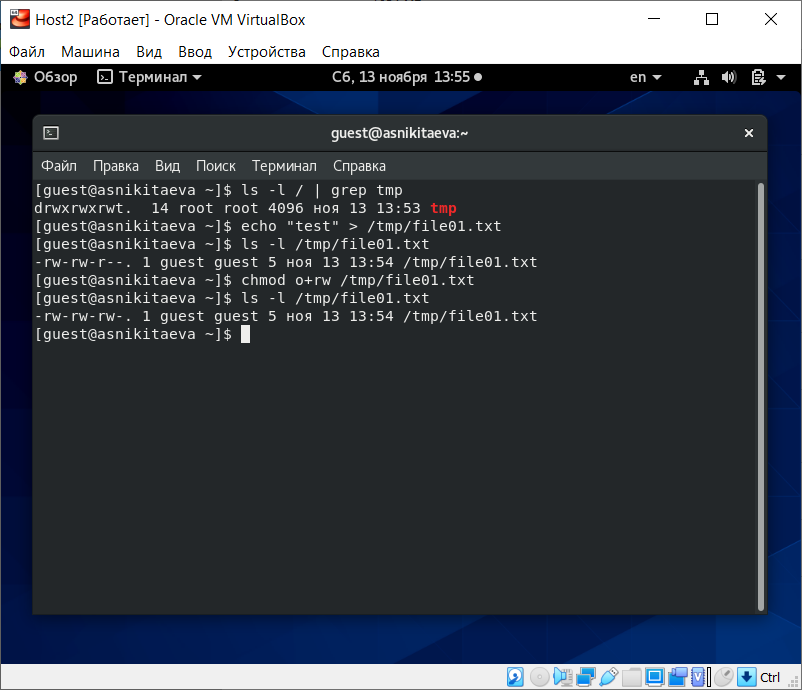


Figure 13: Исследование Sticky-бита от имени guest

2.4. От пользователя guest2 (не являющегося владельцем) попробовала прочитать файл /tmp/file01.txt: cat /tmp/file01.txt. (рис. 14)

2.5. От пользователя guest2 попробовала дозаписать в файл */tmp/file01.txt* слово test2 командой: echo "test2" >> /tmp/file01.txt. (рис. 14) Операция прошла успешно.

2.6. Проверила содержимое файла командой: cat /tmp/file01.txt. (рис. 14)

2.7. От пользователя guest2 попробовала записать в файл */tmp/file01.txt* слово test3, стерев при этом всю имеющуюся в файле информацию командой: echo "test3" > /tmp/file01.txt. (рис. 14) Операция прошла успешно.

2.8. Проверила содержимое файла командой: cat /tmp/file01.txt. (рис. 14)

2.9. От пользователя guest2 попробовала удалить файл */tmp/file01.txt* командой: rm /tmp/fileOl.txt. (рис. 14) Операция была не позволена.

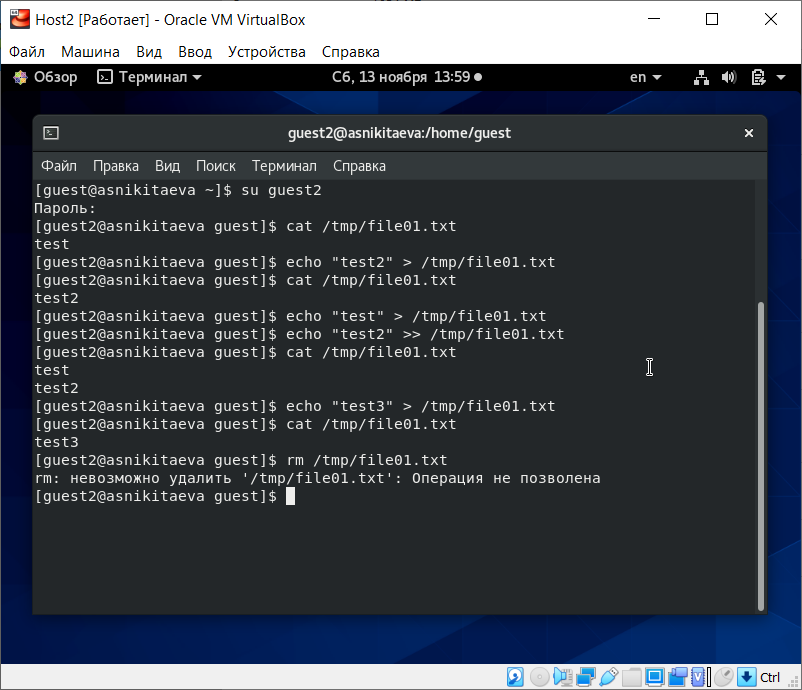


Figure 14: Работа с *file01.txt* от имени guest2 при наличии Sticky-бита

2.10. Повысила свои права до суперпользователя следующей командой: su -, и выполнила после этого команду, снимающую атрибут t (Sticky-бит) с директории */tmp*: chmod -t /tmp. (рис. 15)

2.11. Покинула режим суперпользователя командой: exit. (рис. 15)

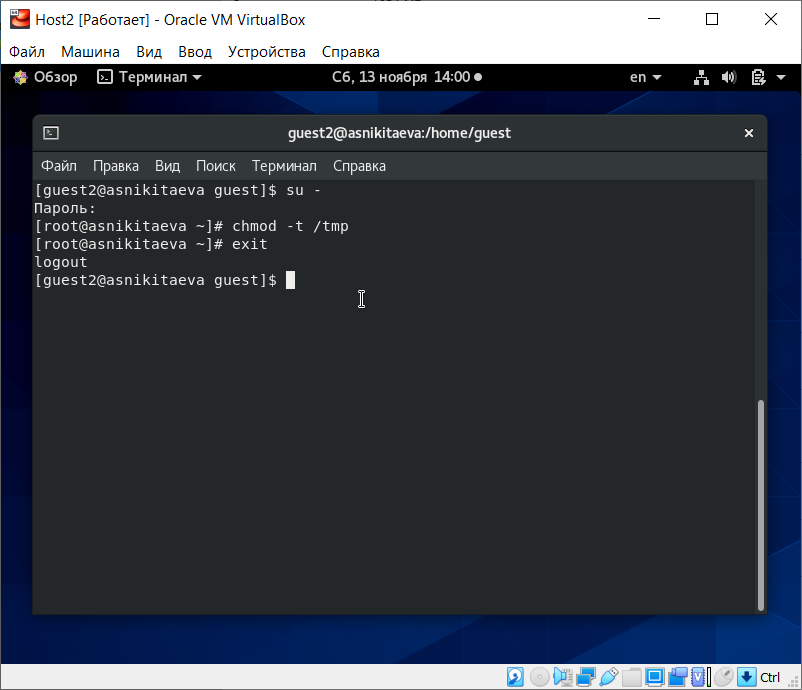


Figure 15: Снятие Sticky-бита с */tmp*

2.12. От пользователя guest2 проверила, что атрибута t у директории */tmp* нет: ls -l / | grep tmp. (рис. 16)

2.13. Повторила предыдущие шаги. (рис. 16) Теперь удалось удалить файл.

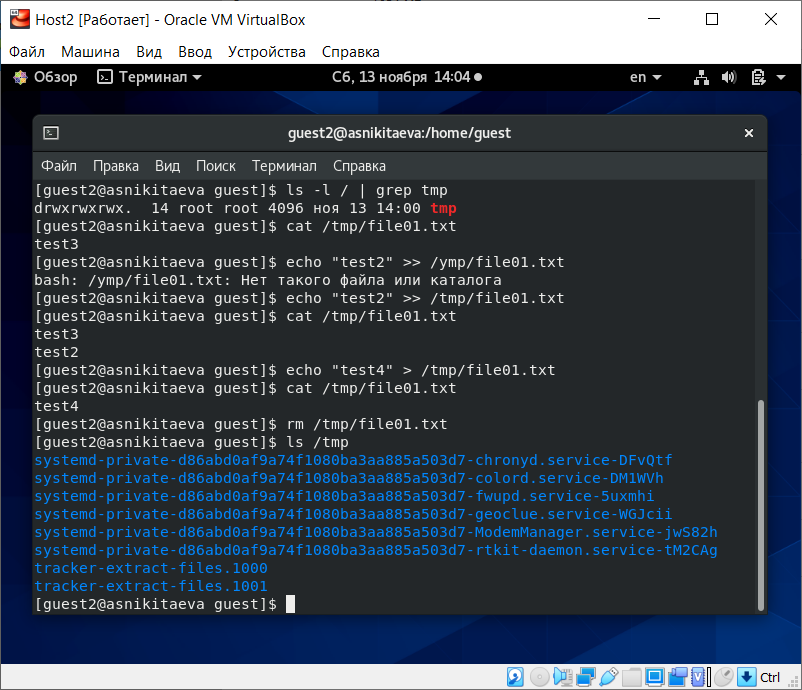


Figure 16: Работа с *file01.txt* от имени guest2 без Sticky-бита

2.14. Да, мне удалось удалить файл от имени пользователя, не являющегося его владельцем.

2.15. Повысила свои права до суперпользователя и вернула атрибут t на директорию /tmp: su -, chmod +t /tmp и exit. (рис. 17)

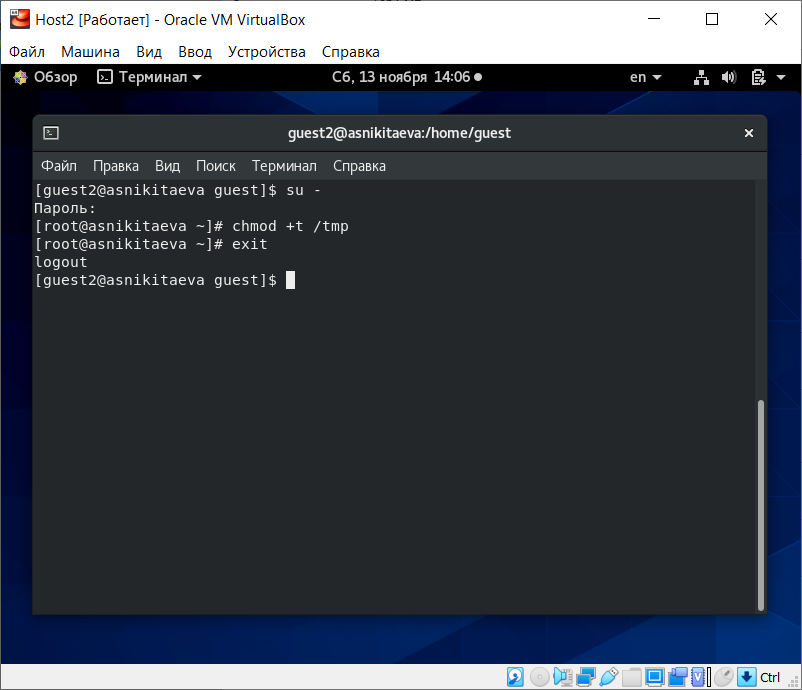


Figure 17: Возвращение Sticky-бита на */tmp*

# Выводы

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

# Список литературы

1. Кулябов Д. С., Королькова А. В., Геворкян М. Н. Информационная безопасность компьютерных сетей. Лабораторная работа № 5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов