Отчёт по лабораторной работе №5

Дисциплина: Информационная безопасность

Выполнил: Танрибергенов Эльдар

Содержание

Список иллюстраций

# 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 Задания

1. Исследовать *SetUID*- и *SetGID*-биты.
2. Исследовать *Sticky*-бит.

# 3 Указания к работе

Подготовка лабораторного стенда. Помимо прав администратора для выполнения части заданий потребуются средства разработки приложений. В частности, при подготовке стенда следует убедиться, что в системе установлен компилятор *gcc*. Так как программы с установленным битом *SetUID* могут представлять большую брешь в системе безопасности, в современных системах используются дополнительные механизмы защиты. Проследите, чтобы система защиты *SELinux* не мешала выполнению заданий работы. Отключите систему запретов до очередной перезагрузки системы командой *setenforce 0*. После этого команда *getenforce* должна выводить *Permissive*.

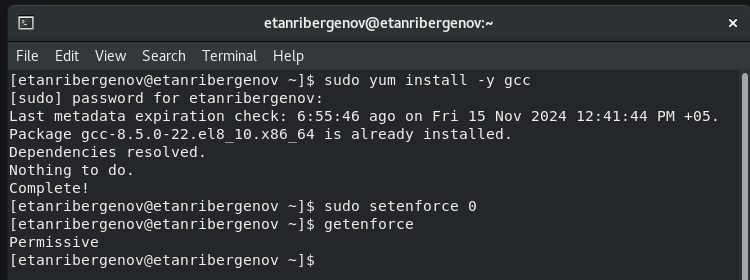


Рис. 1: Подготовка лабораторного стенда

# 4 Выполнение лабораторной работы

1. Исследование *SetUID* и *SetGID*-битов

1.1. Вошёл в систему от имени пользователя *guest-etanribergenov*.

1.2. Создал программу *simpleid.c*

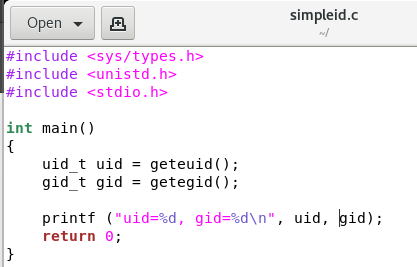


Рис. 2: Программа, выводящая UID и GID

1.3. Скомплилировал программу и убедился, что файл программы создан

gcc simpleid.c -o simpleid

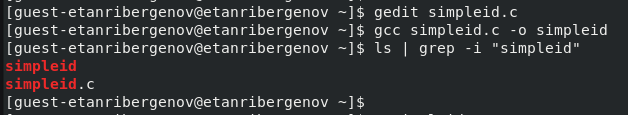


Рис. 3: Компилирование программы

1.4. Выполнил программу *simpleid*:

./simpleid

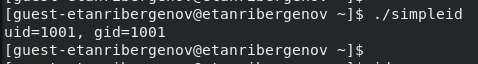


Рис. 4: Выполнение программы

1.5. Выполнил системную программу *id*:

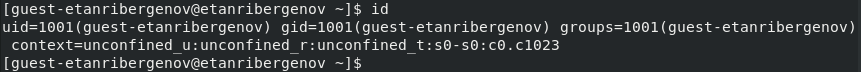


Рис. 5: Системная программа id

Выведенные данные совпадают.

1.6. Усложнил программу, добавив вывод действительных идентификаторов, получившуюся программу назвал *simpleid2.c* :

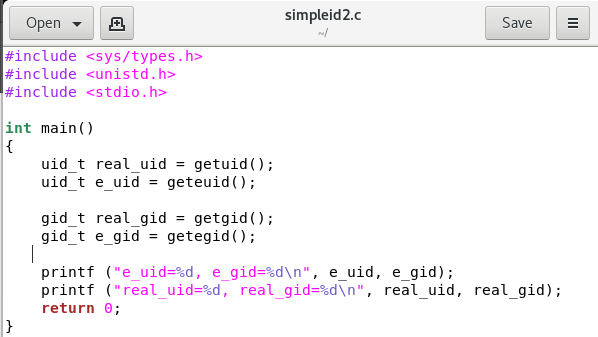


Рис. 6: Добавление в программу действительных идентификаторов

1.7. Скомпилируйте и запустите *simpleid2.c*:

gcc simpleid2.c -o simpleid2

./simpleid2

Рис. 7: Компилирование программы

Рис. 7: Компилирование программы

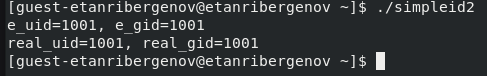


Рис. 8: Запуск программы

1.8. От имени суперпользователя выполните команды:

chown root:guest-etanribergenov /home/guest-etanribergenov/simpleid2

chmod u+s /home/guest-etanribergenov/simpleid2

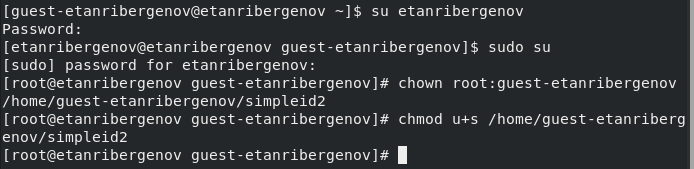


Рис. 9: Смена владельца файла и добавление SetUID-бита

1.9. Использовал *sudo su*, чтобы повысить права пользователя до прав суперпользователя.

1.10. Выполнил проверку правильности установки новых атрибутов и смены владельца файла *simpleid2*:

ls -l simpleid2

Рис. 10: Проверка добавления SUID-бита

Рис. 10: Проверка добавления SUID-бита

1.11. Запустил *simpleid2* и *id*:

./simpleid2

id

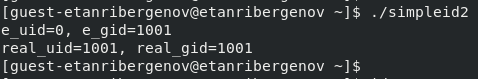


Рис. 11: Запуск программы simpleid2

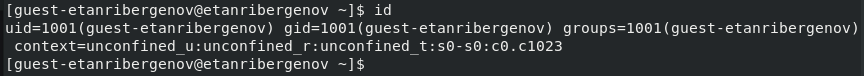


Рис. 12: Запуск системной программы id

Значение *e\_uid* стало 0, в то время, как настоящий идентификатор *uid* и *uid* из программы *id* вывели 1001.

1.12. Проделал тоже самое относительно *SetGID*-бита.

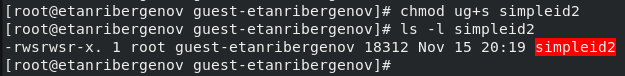


Рис. 13: Добавление программе SetGID-бит

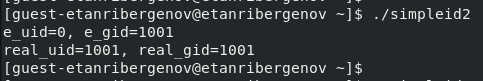


Рис. 14: Запуск программы simpleid2 после добавления SGID-бита

1.13. Создал программу *readfile.c*

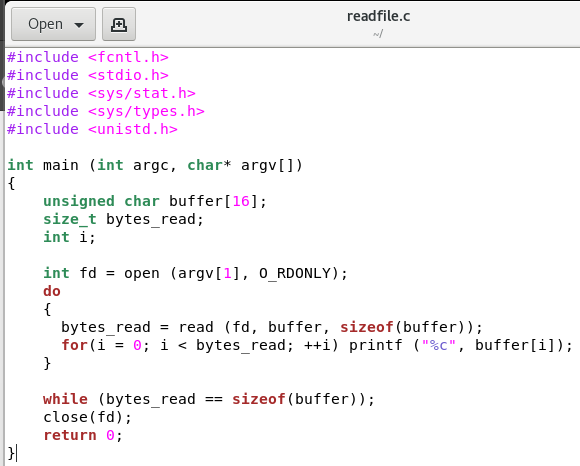


Рис. 15: Программа, считывающая и выводящая в консоль содержимое файла

1.14. Откомпилировал её.

gcc readfile.c -o readfile

Рис. 16: Компилирование программы

Рис. 16: Компилирование программы

1.15. Сменил владельца у файла *readfile.c* и изменил права так, чтобы только суперпользователь (*root*) мог прочитать его, a *guest-etanribergenov* не мог.

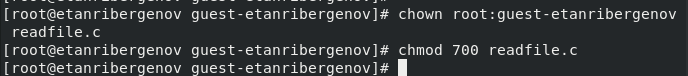


Рис. 17: Смена владельца файла и разрешение только суперпользователю читать его

1.16. Проверил, что пользователь *guest-etanribergenov* не может прочитать файл *readfile.c*.

Рис. 18: Проверка невозможности чтения файла пользователем guest-etanribergenov

Рис. 18: Проверка невозможности чтения файла пользователем guest-etanribergenov

1.17. Сменил у программы *readfile* владельца и установил *SetUID*-бит.

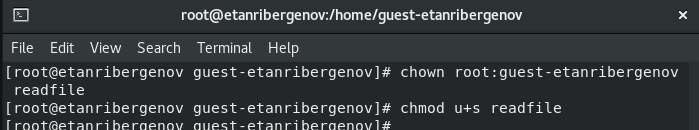


Рис. 19: Смена владельца у программы readfile и установка SetUID-бита

1.18. Проверил, может ли программа *readfile* прочитать файл *readfile.c* - может.

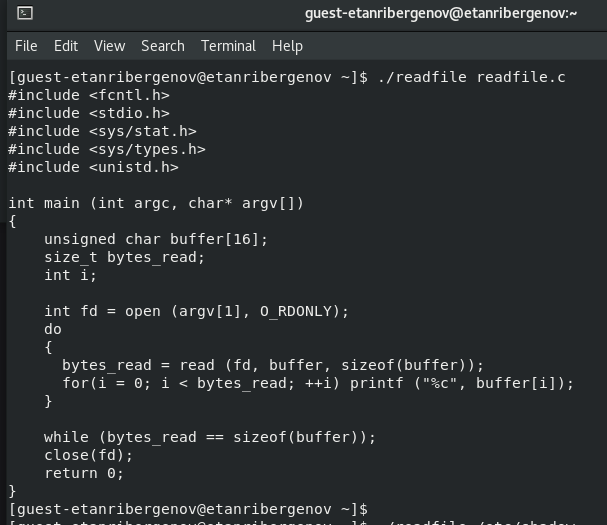


Рис. 20: Проверка возможности чтения программой readfile другого файла

1.19. Проверил, может ли программа *readfile* прочитать файл */etc/shadow* - может.

Это возможно потому, что пользователем программы стал суперпользователь (*root*).

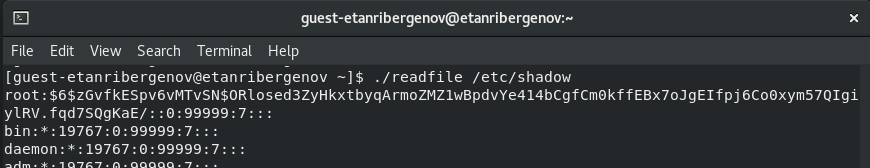


Рис. 21: Проверка возможности чтения программой readfile файла /etc/shadow

1. Исследование Sticky-бита

2.1. Выяснил, установлен ли атрибут *Sticky* на директории */tmp*, для чего выполнил команду

ls -l / | grep tmp

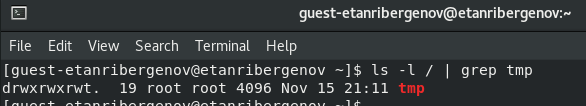


Рис. 22: Проверка наличия Sticky-бита на директории /tmp

2.2. От имени пользователя *guest-etanribergenov* создал файл *file01-etanribergenov.txt* в директории */tmp* со словом test:

echo "test" > /tmp/file01-etanribergenov.txt

Рис. 23: Создание файла со словом «test» в директории /tmp от имени пользователя guest-etanribergenov

Рис. 23: Создание файла со словом «test» в директории /tmp от имени пользователя guest-etanribergenov

2.3. Просмотрел атрибуты у только что созданного файла и разрешил чтение и запись для категории пользователей «все остальные»:

ls -l /tmp/file01-etanribergenov.txt

chmod o+rw /tmp/file01-etanribergenov.txt

ls -l /tmp/file01-etanribergenov.txt

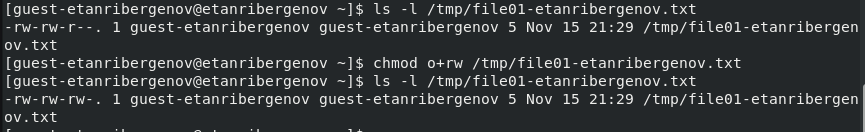


Рис. 24: Просмотр атрибутов и разрешение на чтение и запись для категории пользователей «остальные»

2.4. От пользователя *guest2-etanribergenov* (не являющегося владельцем) попробуйте прочитать файл */tmp/file01-etanribergenov.txt*:

cat /tmp/file01-etanribergenov.txt

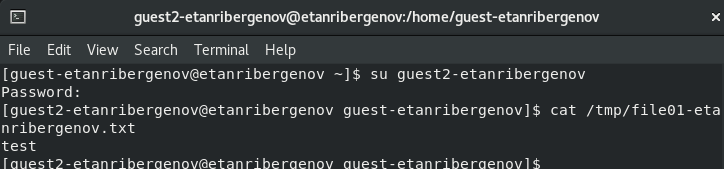


Рис. 25: Попытка чтения файла пользователем, не являющимся его владельцем

2.5. От пользователя *guest2-etanribergenov* попробуйте дозаписать в файл */tmp/file01-etanribergenov.txt* слово «test2» командой

echo "test2" > /tmp/file01-etanribergenov.txt

Рис. 26: Попытка записи в файл пользователем, не являющимся его владельцем

Рис. 26: Попытка записи в файл пользователем, не являющимся его владельцем

Удалось выполнить операцию.

2.6. Проверил содержимое файла командой

cat /tmp/file01-etanribergenov.txt

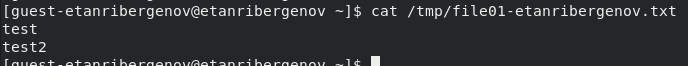


Рис. 27: Проверка содержимого файла

2.7. От пользователя *guest2-etanribergenov* попробуйте записать в файл */tmp/file01-etanribergenov.txt* слово «test3», стерев при этом всю имеющуюся в файле информацию командой

echo "test3" > /tmp/file01-etanribergenov.txt

Рис. 28: Попытка перезаписи файла пользователем, не являющимся его владельцем

Рис. 28: Попытка перезаписи файла пользователем, не являющимся его владельцем

Удалось выполнить операцию.

2.8. Проверил содержимое файла командой

cat /tmp/file01-etanribergenov.txt

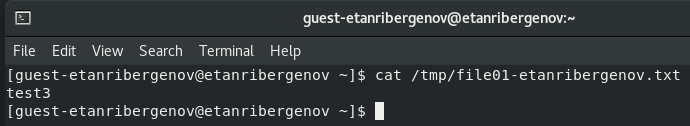


Рис. 29: Проверка содержимого файла

2.9. От пользователя *guest2-etanribergenov* попробуйте удалить файл */tmp/file01-etanribergenov.txt* командой

rm /tmp/fileOl-etanribergenov.txt

Не удалось удалить файл.

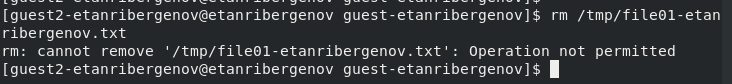


Рис. 30: Попытка удаления файла пользователем, не являющимся его владельцем

2.10. Повысил свои права до суперпользователя следующей командой

su -

и выполнил после этого команду, снимающую атрибут *t* (Sticky-бит) с директории */tmp*:

chmod -t /tmp

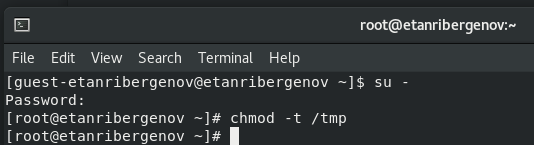


Рис. 31: Снятие Sticky-бита с директории /tmp суперпользователем

2.11. Покинул режим суперпользователя командой

exit

Рис. 32: Выход из режима суперпользователя

Рис. 32: Выход из режима суперпользователя

2.12. От пользователя *guest2-etanribergenov* проверил, что атрибута *t* у директории */tmp* нет:

ls -l / | grep tmp

Рис. 33: Проверка отсутствия Sticky-бита у директории /tmp

Рис. 33: Проверка отсутствия Sticky-бита у директории /tmp

2.13. Повторил предыдущие шаги. Изменений нет - файл всё также можно дополнить и переписать.

2.14. Удалось удалить файл от имени пользователя, не являющегося его владельцем. Это случилось из-за того, что ранее был снят *Sticky*-бит с директории, защищавший файлы от неразрешённых действий.

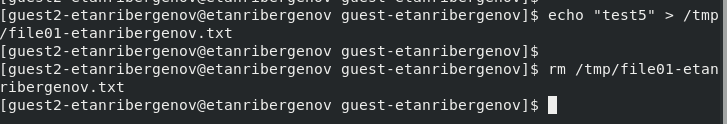


Рис. 34: Попытка удаления файла пользователем, не являющимся его владельцем

2.15. Повысил свои права до суперпользователя и вернул атрибут *t* на директорию */tmp*:

su -  
chmod +t /tmp  
exit

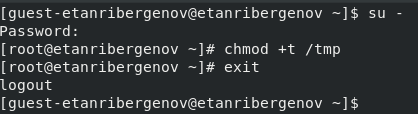


Рис. 35: Возвращение Sticky-бита на директорию /tmp

# 5 Выводы

В результате выполнения работы я изучил механизмы изменения идентификаторов, применения *SetUID*- и *Sticky*-битов. Получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работы механизма смены идентификатора процессов пользователей, а также влияние бита *Sticky* на запись и удаление файлов.