Лабораторная работа №5

Информационная безопасность

Александрова Ульяна Вадимовна

Содержание

# 1 Цель работы

Целью работы является 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

Setuid, Setgid и Sticky Bit - это специальные типы разрешений позволяют задавать расширенные права доступа на файлы или каталоги.

# 3 Подготовка к выполнению работы

Я проверила, установлен ли у меня gcc командой **yum install gcc**. Он установлен и обновлен до последней версии (рис. 1).

![Рис. 1: Проверка установки ПО](data:application/octet-stream;base64,)

Рис. 1: Проверка установки ПО

Помимо этого, я отключила систему запретов до очередной перезагрузки системы командой **setenforce 0**. После этого команда getenforce выводит Permissive (рис. 2).

![Рис. 2: setenforce 0](data:application/octet-stream;base64,)

Рис. 2: setenforce 0

# 4 Выполнение лабораторной работы

## 4.1 Создание программы

Я вошла в систему от имени пользователя guest (рис. 3).

![Рис. 3: Вход в систему от другого пользователя](data:application/octet-stream;base64,)

Рис. 3: Вход в систему от другого пользователя

Далее создала программу simpleid.c и заполнила ее (рис. 4), (рис. 5).

![Рис. 4: Создание программы](data:application/octet-stream;base64,)

Рис. 4: Создание программы

![Рис. 5: Заполнение элементарной программы](data:application/octet-stream;base64,)

Рис. 5: Заполнение элементарной программы

Скомплилировала файл через **gcc simpleid.c -o simpleid** и выполнила программу simpleid (рис. 6).

![Рис. 6: Компиляция и запуск программы](data:application/octet-stream;base64,)

Рис. 6: Компиляция и запуск программы

Выполнила системную программу id. Результаты похожи. Gid и uid одинаковые, однако команда id дает больше информации (рис. 7).

![Рис. 7: Команда id](data:application/octet-stream;base64,)

Рис. 7: Команда id

Усложнила программу, добавив вывод действительных идентификаторов (рис. 8).

![Рис. 8: Заполнение программы](data:application/octet-stream;base64,)

Рис. 8: Заполнение программы

Скомпилировала и запустила simpleid2.c (рис. 9).

![Рис. 9: Компиляция и запуск программы](data:application/octet-stream;base64,)

Рис. 9: Компиляция и запуск программы

От имени суперпользователя выполнила команды **chown root:guest /home/guest/simpleid2, chmod u+s /home/guest/simpleid2** (рис. 10).

![Рис. 10: Поменяла владельца программы](data:application/octet-stream;base64,)

Рис. 10: Поменяла владельца программы

Выполнила проверку правильности установки новых атрибутов и смены владельца файла simpleid2 (рис. 11).

![Рис. 11: ls -l](data:application/octet-stream;base64,)

Рис. 11: ls -l

Запустила simpleid2 и id. Результаты похожи. Gid и uid одинаковые, однако команда id дает больше информации (рис. 12).

![Рис. 12: Сравнение результатов](data:application/octet-stream;base64,)

Рис. 12: Сравнение результатов

Создала программу readfile.c (рис. 13).

![Рис. 13: Создание программы](data:application/octet-stream;base64,)

Рис. 13: Создание программы

Откомпилировала её (рис. 14).

![Рис. 14: Компиляция программы](data:application/octet-stream;base64,)

Рис. 14: Компиляция программы

Сменила владельца у файла readfile.c и изменила права так, чтобы только суперпользователь (root) мог прочитать его, a aleksandrovauv не мог. Проверила, может ли пользователь прочитать файл readfile.c. Не может (рис. 15).

![Рис. 15: Смена владельца](data:application/octet-stream;base64,)

Рис. 15: Смена владельца

Программа readfile в целом может прочитать файл /etc/shadow (рис. 16).

![Рис. 16: Результат работы программы](data:application/octet-stream;base64,)

Рис. 16: Результат работы программы

## 4.2 Исследование Sticky-бита

Выяснила, установлен ли атрибут Sticky на директории /tmp. Установлен.  
От имени пользователя guest создала файл file01.txt в директории /tmp со словом test через **echo “test” > /tmp/file01.txt**. Затем просмотрите атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные» при помощи утилиты **chmod o+rw /tmp/file01.txt** (рис. 17).

![Рис. 17: Выполнение задач](data:application/octet-stream;base64,)

Рис. 17: Выполнение задач

От пользователя guest2 (не являющегося владельцем) попробовала прочитать файл /tmp/file01.txt. Попробовала дозаписать в файл /tmp/file01.txt слово test2 командой **echo “test2” > /tmp/file01.txt**. Мне отказано в доступе. То же самое попробовала сделать с test3, но мне снова отказано в доступе. Файл не записался (рис. 18).

![Рис. 18: Работа с файлами](data:application/octet-stream;base64,)

Рис. 18: Работа с файлами

От пользователя guest2 попробовала удалить файл /tmp/file01.txt. Мне отказали в доступе. Потом я повысила свои права до суперпользователя командой **su -** и сняла атрибут t (Sticky-бит) с директории /tmp. От пользователя guest2 проверила, что атрибута t у директории /tmp нет. Повторила предыдущие шаги. Файл удалился (рис. 19).

![Рис. 19: Работа с атрибутом t](data:application/octet-stream;base64,)

Рис. 19: Работа с атрибутом t

Вернула атрибут t (рис. 20).

![Рис. 20: Возвращение Sticky](data:application/octet-stream;base64,)

Рис. 20: Возвращение Sticky

# 5 Выводы

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Полученила практических навыков работы в консоли с дополнительными атрибутами.