

# **Лабораторная работа №5**

**Дисциплина: Основы информационной безопасности**

Коновалова Татьяна Борисовна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Теоретические данные</b>	<b>6</b>
<b>3</b>	<b>Задание</b>	<b>8</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>9</b>
4.1	Создание программы . . . . .	9
4.2	Исследование Sticky-бита . . . . .	16
<b>5</b>	<b>Выводы</b>	<b>18</b>
<b>6</b>	<b>Библиография</b>	<b>19</b>

## Список иллюстраций

4.1	Проверка компилятора gcc . . . . .	9
4.2	Вывод программой “Permissive” . . . . .	10
4.3	Проверка команд команд “whereis gcc” и “whereis g++” . . . . .	10
4.4	Создание программы simpleid.c . . . . .	11
4.5	Текст программы simpleid.c . . . . .	11
4.6	Компиляция и запуск simpleid . . . . .	11
4.7	Создание файла для программы simpleid2.c . . . . .	12
4.8	Текст программы simpleid2.c . . . . .	12
4.9	Запуск программы simpleid2.c . . . . .	13
4.10	Смена владельца и установка SetUID . . . . .	13
4.11	Запуск simpleid2 . . . . .	13
4.12	SetGID-бит . . . . .	13
4.13	Текст программы readfile.c . . . . .	14
4.14	Смена владельца и прав доступа у файла readfile.c . . . . .	14
4.15	Ошибка при прочтении файла readfile.c . . . . .	14
4.16	Меняем владельца файла readfile . . . . .	15
4.17	Убедилась, что мы можем читать файл readfile . . . . .	15
4.18	Чтение файла /etc/shadow . . . . .	15
4.19	Создание файла file01.txt . . . . .	16
4.20	Действия над file01.txt от лица guest2 . . . . .	16
4.21	Удаление Sticky-бита . . . . .	17
4.22	Повтор действий . . . . .	17
4.23	Возвращение Sticky-бита . . . . .	17

## **Список таблиц**

# 1 Цель работы

Цель лабораторной работы — Изучить механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получить практические навыки работы в консоли с дополнительными атрибутами. Рассмотреть работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

## 2 Теоретические данные

Типы разрешений:

SetUID, SetGID и Sticky — это специальные типы разрешений, которые позволяют задавать расширенные права доступа на файлы и каталоги.

- SetUID — это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла. Другими словами, использование этого бита позволят поднять привилегии пользователя в случае, если это необходимо. Наличие SetUID бита выражается в том, что на месте классического бита x выставлен специальный бит s: -rwsr-xr-x
- SetGID — очень похож на SetUID с отличием, что файл будет запускаться от имени группы, который владеет файлом: -rwxr-sr-x
- Sticky — в случае, если этот бит установлен для папки, то файлы в этой папке могут быть удалены только их владельцем. Наличие этого бита показывается через букву t в конце всех прав: drwxrwxrwx t

Атрибуты — это набор основных девяти битов, определяющих какие из пользователей обладают правами на чтение, запись и исполнение. Первые три бита отвечают права доступа владельца, вторые — для группы пользователей, последнее — для всех остальных пользователей в системе.

Установка атрибутов производится командой `chmod`. Установка бита чтения (r) позволяет сделать файл доступным для чтения. Наличие бита записи (w) позволяет изменять файл. Установка бита запуска (x) позволяет запускать файл на исполнение.

Расширенные атрибуты — это система дополнительной информации, которая может быть добавлена к файлу или директории в файловой системе.

Некоторые примеры расширенных атрибутов:

- `a` — файл можно открыть только в режиме добавления.
- `A` — при доступе к файлу его запись `atime` не изменяется.
- `s` — файл автоматически сжимается.
- `e` — файл использует экстенды.
- `E` — файл, каталог или символьная ссылка зашифрованы файловой системой.
- `F` — поиски путей в директории выполняются без учёта регистра.
- `i` — файл не может быть изменён.
- `m` — файл не сжимается.

Установка атрибутов производится командой `chmod`. Установка бита чтения (`r`) позволяет сделать файл доступным для чтения. Наличие бита записи (`w`) позволяет изменять файл. Установка бита запуска (`x`) позволяет запускать файл на исполнение.

## 3 Задание

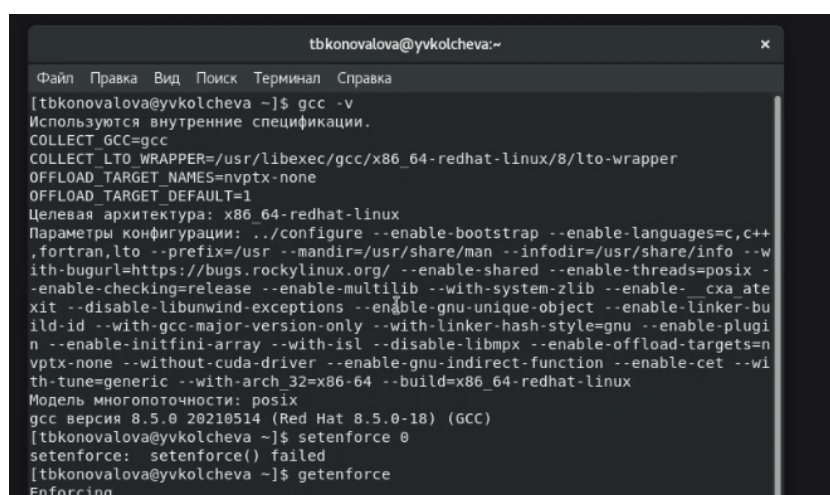
1.Создание и работа с программой simpleid.c ; 2.Исследование Sticky-бита.



## 4 Выполнение лабораторной работы

### 4.1 Создание программы

1). Убедилась, то компилятор gcc установлен, используя команду “gcc -v”. Затем отключила систему запретов до очередной перезагрузки системы командой “sudo setenforce 0”, после чего команда “getenforce” вывела “Permissive” (рис. [4.1]) и (рис. [4.2]).



```
tbkonovalova@yvkolcheva:~  
Файл Правка Вид Поиск Терминал Справка  
[tbkonovalova@yvkolcheva ~]$ gcc -v  
Используются внутренние спецификации.  
COLLECT_GCC=gcc  
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/8/lto-wrapper  
OFFLOAD_TARGET_NAMES=nvptx-none  
OFFLOAD_TARGET_DEFAULT=1  
Целевая архитектура: x86_64-redhat-linux  
Параметры конфигурации: ../configure --enable-bootstrap --enable-languages=c,c++  
 ,fortran,lto --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --w  
ith-bugurl=https://bugs.rockylinux.org/ --enable-shared --enable-threads=posix -  
-enable-checking=release --enable-multilib --with-system-zlib --enable-__cxa_at  
exit --disable-libunwind-exceptions --enable-gnu-unique-object --enable-linker-bu  
ild-id --with-gcc-major-version-only --with-linker-hash-style=gnu --enable-plugi  
n --enable-initfini-array --with-isl --disable-libmpx --enable-offload-targets=n  
vptx-none --without-cuda-driver --enable-gnu-indirect-function --enable-cet --wi  
th-tune=generic --with-arch_32=x86_64 --build=x86_64-redhat-linux  
Модель многопоточности: posix  
gcc версия 8.5.0 20210514 (Red Hat 8.5.0-18) (GCC)  
[tbkonovalova@yvkolcheva ~]$ setenforce 0  
setenforce: setenforce() failed  
[tbkonovalova@yvkolcheva ~]$ getenforce  
Enforcing
```

Рис. 4.1: Проверка компилятора gcc

```
tbkonovalova@yvkolcheva:/home/tbkonovalova
Файл Правка Вид Поиск Терминал Справка
[tbkonovalova@yvkolcheva ~]$ getenforce
Enforcing
[tbkonovalova@yvkolcheva ~]$ setenforce0
bash: setenforce0: команда не найдена...
Аналогичная команда: 'setenforce'
[tbkonovalova@yvkolcheva ~]$ setenforce0
bash: setenforce0: команда не найдена...
Аналогичная команда: 'setenforce'
[tbkonovalova@yvkolcheva ~]$ setenforce 0
setenforce: setenforce() failed
[tbkonovalova@yvkolcheva ~]$ sudo setenforce 0
[sudo] пароль для tbkonovalova:
tbkonovalova is not in the sudoers file. This incident will be reported.
[tbkonovalova@yvkolcheva ~]$ sudo setenforce 0
[sudo] пароль для tbkonovalova:
Попроуйте ещё раз.
[sudo] пароль для tbkonovalova:
tbkonovalova is not in the sudoers file. This incident will be reported.
[tbkonovalova@yvkolcheva ~]$ su
Пароль:
[root@yvkolcheva tbkonovalova]# setenforce 0
[root@yvkolcheva tbkonovalova]# getenforce
Permissive
[root@yvkolcheva tbkonovalova]#
```

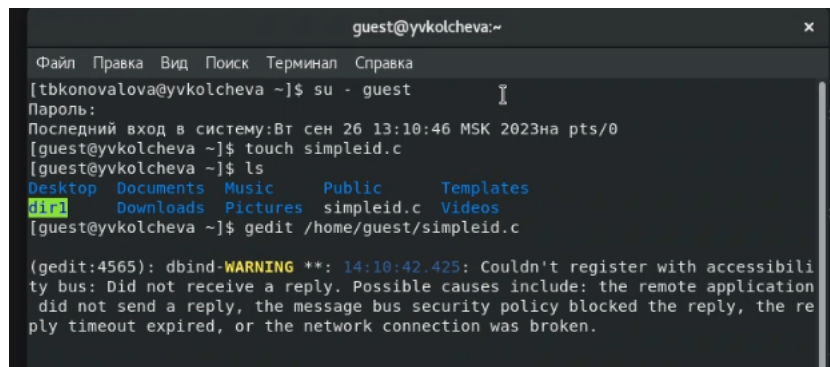
Рис. 4.2: Вывод программой “Permissive”

Проверила успешное выполнение команд “whereis gcc” и “whereis g++” (их расположение) (рис. [4.3]).

```
Permissive
[root@yvkolcheva tbkonovalova]# whereis gcc
gcc: /usr/bin/gcc /usr/lib/gcc /usr/libexec/gcc /usr/share/man/man1/gcc.1.gz /usr/share/info/gcc.info.gz
[root@yvkolcheva tbkonovalova]# whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
[root@yvkolcheva tbkonovalova]#
```

Рис. 4.3: Проверка команд команд “whereis gcc” и “whereis g++”

2). Вошла в систему от имени пользователя guest командой “su - guest”. Создала программу simpleid.c командой “touch simpleid.c” и открыла её в редакторе командой “gedit /home/guest/simpleid.c” (рис. [4.4]).



```
guest@yvkolcheva:~  
Файл Правка Вид Поиск Терминал Справка  
[tbkonovalova@yvkolcheva ~]$ su - guest  
Пароль:  
Последний вход в систему: Вт сен 26 13:10:46 MSK 2023 на pts/0  
[guest@yvkolcheva ~]$ touch simpleid.c  
[guest@yvkolcheva ~]$ ls  
Desktop Documents Music Public Templates  
dir Downloads Pictures simpleid.c Videos  
[guest@yvkolcheva ~]$ gedit /home/guest/simpleid.c  
  
(gedit:4565): dbind-WARNING **: 14:10:42.425: Couldn't register with accessibility bus: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.
```

Рис. 4.4: Создание программы simpleid.c

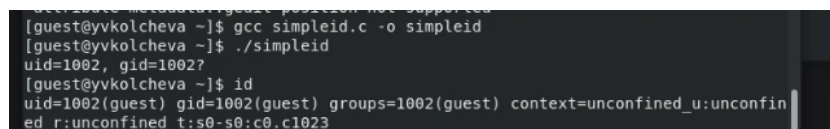
Создала программу simpleid.c со следующим текстом (рис. [4.5]).



```
Open simpleid.c Save  
#include <sys/types.h>  
#include <unistd.h>  
#include <stdio.h>  
  
int  
main ()  
{  
    uid_t uid = geteuid();  
    gid_t gid = getegid();  
    printf("uid=%d, gid=%d\n", uid, gid);  
    return 0;  
}
```

Рис. 4.5: Текст программы simpleid.c

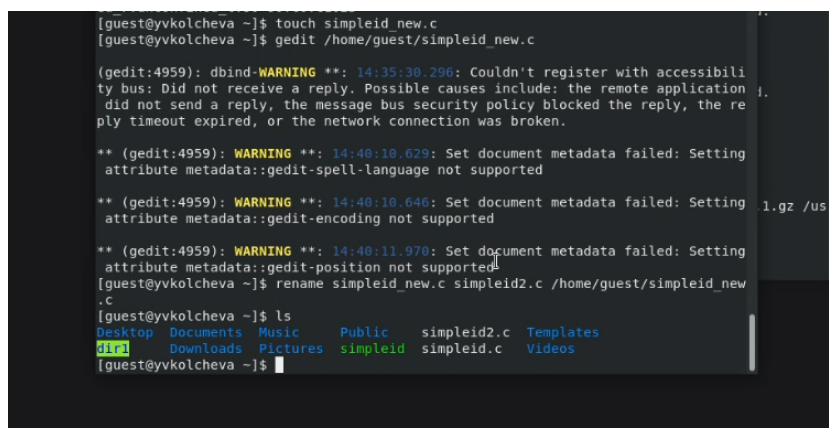
3). Скомпилировала программу с помощью команды gcc и убедилась, что файл действительно создан. Далее запустила исполняемый файл через ./ . Вывод написанной программы совпадает с выводом команды id (рис [4.6]).



```
[guest@yvkolcheva ~]$ gcc simpleid.c -o simpleid  
[guest@yvkolcheva ~]$ ./simpleid  
uid=1002, gid=1002?  
[guest@yvkolcheva ~]$ id  
uid=1002(guest) gid=1002(guest) groups=1002(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Рис. 4.6: Компиляция и запуск simpleid

4). Создание файла для программы simpleid2.c и запуск данного файла для ввода программы (рис. [4.7]).

A terminal window showing the creation of a file named simpleid2.c. The user runs 'touch simpleid2.c' and 'gedit /home/guest/simpleid2.c'. The gedit window shows several warning messages about dbus and metadata. The user then runs 'ls' and the output shows 'simpleid2.c' in the current directory.

```
[guest@yvkolcheva ~]$ touch simpleid2.c
[guest@yvkolcheva ~]$ gedit /home/guest/simpleid2.c

(gedit:4959): dbind-WARNING **: 14:35:30.296: Couldn't register with accessibility bus: Did not receive a reply. Possible causes include: the remote application did not send a reply, the message bus security policy blocked the reply, the reply timeout expired, or the network connection was broken.

** (gedit:4959): WARNING **: 14:40:10.629: Set document metadata failed: Setting attribute metadata::gedit-spell-language not supported

** (gedit:4959): WARNING **: 14:40:10.646: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported

** (gedit:4959): WARNING **: 14:40:11.970: Set document metadata failed: Setting attribute metadata::gedit-position not supported
[guest@yvkolcheva ~]$ rename simpleid2.c simpleid2.c /home/guest/simpleid2.c
[guest@yvkolcheva ~]$ ls
Desktop  Documents  Music      Public    simpleid2.c  Templates
Downloads  Pictures  simpleid  simpleid.c  Videos
```

Рис. 4.7: Создание файла для программы simpleid2.c

Текст усложнённой программы, назвала её simpleid2.c (рис. [4.8]).

A screenshot of a text editor window titled 'simpleid2.c'. The code is a C program that includes <sys/types.h>, <unistd.h>, and <stdio.h>. It defines a main function that gets the real and effective user and group IDs and prints them out.

```
файл машина вид ввод устройства справка
Обзор
simpleid2.c
Open Save
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>

int
main ()
{
    uid_t real_uid = getuid ();
    uid_t e_uid = geteuid ();

    gid_t real_gid = getgid ();
    gid_t e_gid = getegid ();

    printf("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
    printf("real_uid=%d, real_gid=%d\n", real_uid, real_gid);

    return 0;
}
```

Рис. 4.8: Текст программы simpleid2.c

Скомпилировала вторую программу с помощью команды gcc и убедилась, что файл действительно создан. Далее запустила исполняемый файл через ./ . Вывод написанной программы совпадает с выводом команды id (рис [4.9]).

```
[guest@yvvkolcheva ~]$ gcc simpleid2.c -o simpleid2
[guest@yvvkolcheva ~]$ ./simpleid2
e uid=1002, e gid=1002
real uid=1002, real gid=1002
[guest@yvvkolcheva ~]$
```

Рис. 4.9: Запуск программы simpleid2.c

5). От имени суперпользователя сменила владельца файла simpleid2 на root и установила SetUID-бит. После этого через команду `ls -l` убедилась, что бит установлен корректно (рис. [4.16])

```
[guest@yvvkolcheva ~]$ su
Password:
[root@yvvkolcheva guest]# chown root:guest /home/guest/simpleid2
[root@yvvkolcheva guest]# chmod u+s /home/guest/simpleid2
[root@yvvkolcheva guest]# ls -l /home/guest/simpleid2
-rwsrwxr-x. 1 root guest 18312 Oct  2 14:43 /home/guest/simpleid2
[root@yvvkolcheva guest]#
```

Рис. 4.10: Смена владельца и установка SetUID

6). Запустила программу simpleid2 и команду `id`. Теперь вижу, что появились отличия в `uid` строках (рис. [4.11]).

```
[tbkonovalova@yvvkolcheva ~]$ su - guest
Пароль:
Последний вход в систему:Пн окт  2 14:09:30 MSK 2023на pts/1
[guest@yvvkolcheva ~]$ ./simpleid2
e uid=0, e gid=1002
real uid=1002, real gid=1002
[guest@yvvkolcheva ~]$ id
uid=1002(guest) gid=1002(guest) groups=1002(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0-c1023
[guest@yvvkolcheva ~]$
```

Рис. 4.11: Запуск simpleid2

Проделаю выше описанные действия для SetGID-бита. Теперь после запуска simpleid2 увидела отличие и в `gid` строках (рис. [4.12]).

```
[guest@yvvkolcheva ~]$ su
Password:
[root@yvvkolcheva guest]# chown root:guest /home/guest/simpleid2
[root@yvvkolcheva guest]# chmod u+s /home/guest/simpleid2
[root@yvvkolcheva guest]# ls -l /home/guest/simpleid2
-rwsrwxr-x. 1 root guest 18312 Oct  2 14:43 /home/guest/simpleid2
[root@yvvkolcheva guest]# chown root:guest /home/guest/simpleid2
[root@yvvkolcheva guest]# chmod g+s /home/guest/simpleid2
[root@yvvkolcheva guest]#
```

```
[guest@yvvkolcheva ~]$ ./simpleid2
e uid=1002, e gid=1002
real uid=1002, real gid=1002
[guest@yvvkolcheva ~]$ id
uid=1002(guest) gid=1002(guest) groups=1002(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0-c1023
[guest@yvvkolcheva ~]$
```

Рис. 4.12: SetGID-бит

8). Создала программу readfile.c (рис. [4.13]).

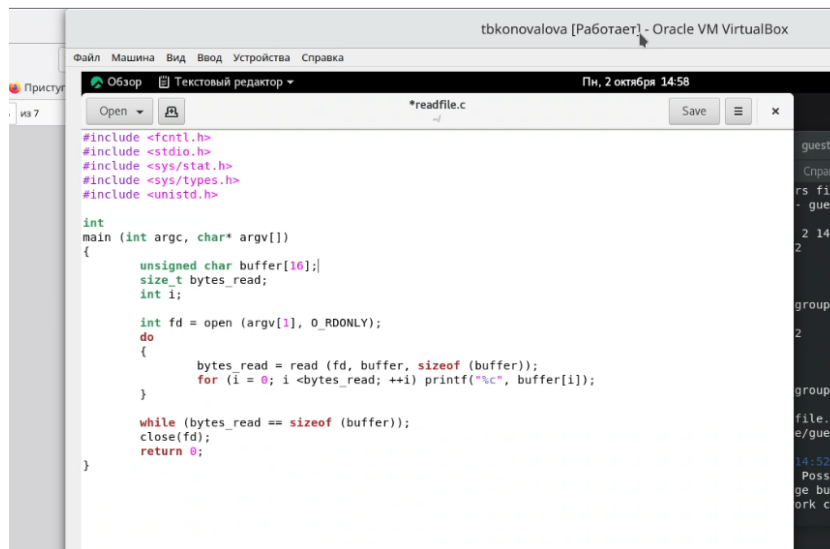


Рис. 4.13: Текст программы readfile.c

Откомпилировала эту программу командой `gcc`. После этого изменила владельца файла `readfile.c` и убрала у пользователя `guest` право на чтение. При попытке прочитать файл от имени пользователя `guest` теперь возникает ошибка (рис. [4.14]) и (рис. [4.15]).

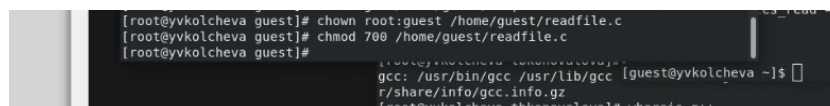


Рис. 4.14: Смена владельца и прав доступа у файла readfile.c



Рис. 4.15: Ошибка при прочтении файла readfile.c

10). Поменяла владельца файла `readfile` и установила на него SetUID-бит (рис. [??]). Запустила исполняемый файл и убедилась, что программа может прочитать файлы `readfile.c` и `/etc/shadow` (рис. [4.17]) и (рис. [4.18]).





## 4.2 Исследование Sticky-бита

1). Выполняя команду `ls -l` выяснила, что на каталоге `/tmp` установлен Sticky-бит. Это видно, т.к. в конце написана `t`. Далее от имени пользователя `guest` создала файл `/tmp/file01.txt`. После этого просмотрела атрибуты только что созданного файла и разрешила всем пользователям право на чтение и запись (рис. [4.19]).

```
[guest@yvkolcheva ~]$ echo "test" > /tmp/file01.txt
[guest@yvkolcheva ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 Oct  2 15:09 /tmp/file01.txt
[guest@yvkolcheva ~]$ chmod o+rw /tmp/file01.txt
[guest@yvkolcheva ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 Oct  2 15:09 /tmp/file01.txt
[guest@yvkolcheva ~]$
```

tbkonovalova@yvkolcheva:~

Файл Правка Вид Поиск Терминал Справка

```
[tbkonovalova@yvkolcheva ~]$ ls -l / | grep tmp
drwxrwxrwt. 15 root root 4096 окт  2 15:01 tmp
[tbkonovalova@yvkolcheva ~]$
```

Рис. 4.19: Создание файла file01.txt

2). От имени пользователя `guest2` прочитала файл `file01.txt` командой `cat`. Далее успешно дозаписала в конец файла строку “test2”, а затем успешно перезаписала содержимое, меняя его на строку “test3”. Однако при попытке удалить файл возникла ошибка (рис. [4.20]).

```
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test
[guest2@yvkolcheva ~]$ echo "test2" >> /tmp/file01.txt
[guest2@yvkolcheva ~]$ echo "test2" > /tmp/file01.txt
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test2
[guest2@yvkolcheva ~]$ echo "test3" > /tmp/file01.txt
[guest2@yvkolcheva ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 6 Oct  2 15:22 /tmp/file01.txt
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test3
[guest2@yvkolcheva ~]$ echo "test2" > /tmp/file01.txt
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test2
[guest2@yvkolcheva ~]$ echo "test3" > /tmp/file01.txt
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test3
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': Operation not permitted
[guest2@yvkolcheva ~]$
```

[guest@yvkolcheva ~]\$ chmod o+rw /tmp/file01.txt
[guest@yvkolcheva ~]\$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 Oct 2 15:09 /tmp/file01.txt
[guest@yvkolcheva ~]\$ chmod g+rw /tmp/file01.txt
[guest@yvkolcheva ~]\$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 6 Oct 2 15:19 /tmp/file01.txt
[guest@yvkolcheva ~]\$

Рис. 4.20: Действия над file01.txt от лица guest2



3). Временно повысила права до суперпользователя и сняла с директории /tmp Sticky-бит. Вышла из режима суперпользователя командой exit (рис. [4.21]).

```
[root@yvkolcheva guest]# chmod -t /tmp
[root@yvkolcheva guest]# exit
exit
[guest@yvkolcheva ~]$
```

Рис. 4.21: Удаление Sticky-бита

3). Убедилась с помощью команды ls -l, что Sticky-бит действительно отсутствует. После этого повторила действия от имени пользователя guest2, описанные выше. В этот раз мне удалось удалить файл file01.txt даже при условии, что guest2 не является его владельцем (рис. [4.22]).

```
[guest2@yvkolcheva ~]$ ls -l / | grep tmp
drwxrwxrwx. 15 root root 4096 Oct 2 15:09 tmp
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test3
[guest2@yvkolcheva ~]$ echo "test2" >> /tmp/file01.txt
[guest2@yvkolcheva ~]$ cat /tmp/file01.txt
test3
test2
[guest2@yvkolcheva ~]$ rm /tmp/file01.txt
[guest2@yvkolcheva ~]$
```

Рис. 4.22: Повтор действий

4). Временно повысила права до суперпользователя и вернула Sticky-бит на каталог /tmp (рис. [4.23]).

```
[guest2@yvkolcheva ~]$ ls -l / | grep tmp
drwxrwxrwx. 15 root root 4096 Oct 2 15:26 tmp
[guest2@yvkolcheva ~]$
```

```
-rw-rw-r--. root root 4096 Oct 2 15:26 tmp
[guest@yvkolcheva guest]# chmod +t /tmp
[guest@yvkolcheva guest]# exit
exit
[guest@yvkolcheva ~]$
```

Рис. 4.23: Возвращение Sticky-бита

## 5 Выводы

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

## 6 Библиография

### СПИСОК ЛИТЕРАТУРЫ

- 1.Медведовский И.Д., Семьянов П.В., Платонов В.В. Атака через Internet. — НПО “Мир и семья-95”, 1997. — URL: <http://bugtraq.ru/library/books/attack1/index.html>
- 2.Теоретические знания, приведённые в Лабораторной работе №5 - [https://esystem.rudn.ru/pluginfile.php/2090129/mod\\_resource/content/2/005-lab\\_discret\\_sticky.pdf](https://esystem.rudn.ru/pluginfile.php/2090129/mod_resource/content/2/005-lab_discret_sticky.pdf)
- 3.Запечников С. В. и др. Информационная безопасность открытых систем. Том 1. — М.: Горячая линия -Телеком, 2006.

### СПИСОК ИНТЕРНЕТ-ИСТОЧНИКОВ

- 1.[Электронный ресурс] - доступ: <https://codeby.school/blog/informacionnaya-bezopasnost/razgranichenie-dostupa-v-linux-znakomstvo-s-astra-linux>
- 2.[Электронный ресурс] - доступ: <https://debianinstall.ru/diskretnoe-razgranichenie-dostupa-linux/>