

Отчёта по лабораторной работе №5

**Дискреционное разграничение прав в Linux. Исследование влияния
дополнительных атрибутов**

Акондзо Жордани Лади Гаэл

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Часть 1: Изучение SetUID- и SetGID-битов	6
2.1.1	Вопросы и Решение	6
2.2	Часть 2: Исследование Sticky-бита	14
2.2.1	Вопросы и Решение	14
3	Выводы	19

Список иллюстраций

2.1	Вход в систему	6
2.2	Создание программы simpleid.c и её компиляция	7
2.3	Создание программы simpleid.c и её компиляция	7
2.4	Запуск программы	7
2.5	Сравнение результата с системной командой id	8
2.6	Усложнение программы, создание simpleid2.c	8
2.7	Усложнение программы, создание simpleid2.c	8
2.8	Усложнение программы, создание simpleid2.c	9
2.9	Изменение владельца программы	9
2.10	Изменение владельца программы	9
2.11	Запуск simpleid2 и id для сравнения результатов	10
2.12	Запуск simpleid2 и id для сравнения результатов	10
2.13	Применение SetGID-бита к simpleid2	11
2.14	Создание программы readfile.c для чтения файла	11
2.15	Создание программы readfile.c для чтения файла	12
2.16	Изменение владельца файла и прав доступа	12
2.17	Проверка прочтения	12
2.18	Установка SetUID-бита на программу readfile	13
2.19	Проверка, может ли программа readfile прочитать файл readfile.c	13
2.20	Проверка, может ли программа readfile прочитать файл /etc/shadow	14
2.21	Проверка наличия Sticky-бита на директории /tmp	14
2.22	Создание файла в /tmp и изменение прав доступа	15
2.23	Создание файла в /tmp и изменение прав доступа	15
2.24	Чтение и изменение файла другим пользователем guest2	15
2.25	Чтение и изменение файла другим пользователем guest2	16
2.26	Чтение и изменение файла другим пользователем guest2	16
2.27	Чтение и изменение файла другим пользователем guest2	16
2.28	Чтение и изменение файла другим пользователем guest2	16
2.29	Чтение и изменение файла другим пользователем guest2	17
2.30	Удаление Sticky-бита с /tmp	17
2.31	Проверка атрибута	17
2.32	Возвращение Sticky-бита на /tmp	18
2.33	Повторение действия	18

List of Tables

1 Цель работы

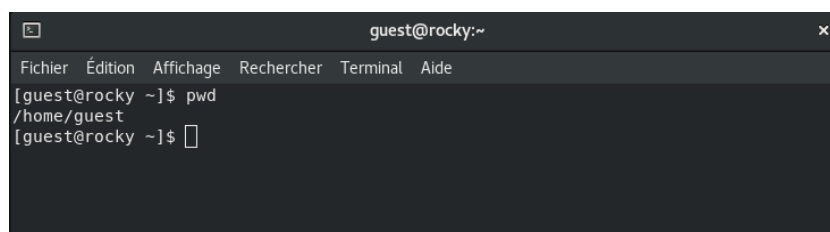
- Изучение механизмов изменения идентификаторов, применение SetUID- и SetGID-битов.
- Изучение действия Sticky-бита на запись и удаление файлов в общей директории.

2 Выполнение лабораторной работы

2.1 Часть 1: Изучение SetUID- и SetGID-битов

2.1.1 Вопросы и Решение

1. Вошёл в систему от имени пользователя guest (рис. 2.1).



```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ pwd  
/home/guest  
[guest@rocky ~]$
```

Рис. 2.1: Вход в систему

2. Создание программы `simpleid.c` и её компиляция:

- Программа `simpleid.c` выводит эффективный идентификатор пользователя (UID) и группы (GID). Она компилируется командой: (рис. 2.2) и (рис. 2.3)



```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.8 simpleid.c  
  
#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. 2.2: Создание программы simpleid.c и её компиляция

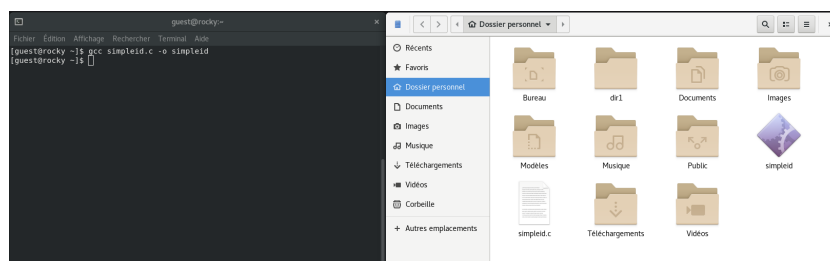
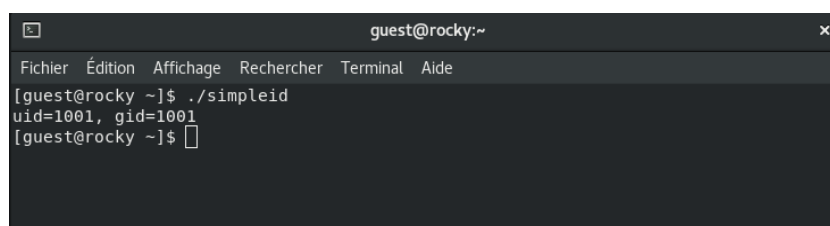


Рис. 2.3: Создание программы simpleid.c и её компиляция

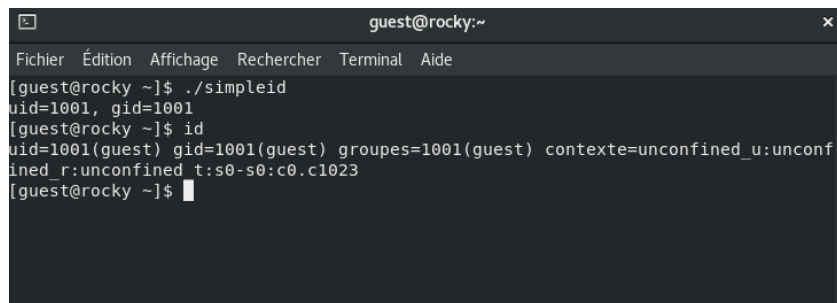
3. Запуск программы simpleid и сравнение результата с системной командой id:

- Программа simpleid и команда id показывают текущий эффективный UID и GID пользователя, что должно совпадать. (рис. 2.4) и (рис. 2.5)



```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@rocky ~]$
```

Рис. 2.4: Запуск программы

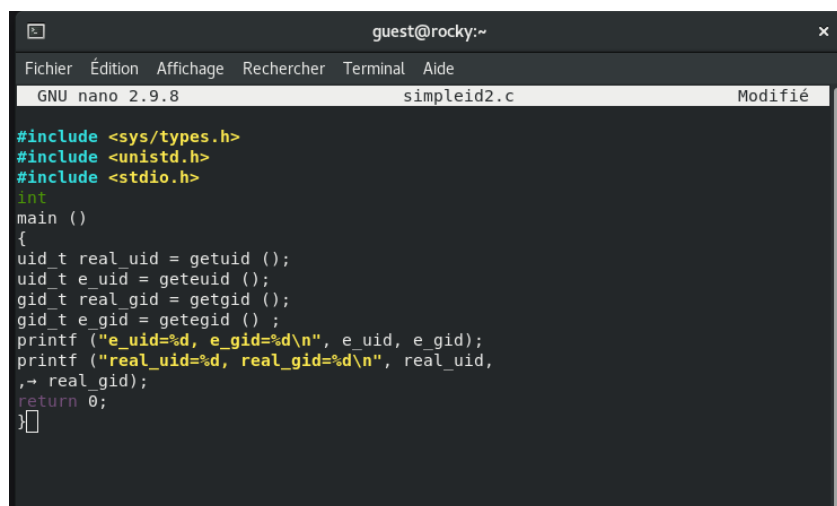


```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@rocky ~]$ id  
uid=1001(guest) gid=1001(guest) groupes=1001(guest) contexte=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@rocky ~]$
```

Рис. 2.5: Сравнение результата с системной командой id

4. Усложнение программы, создание simpleid2.c:

- Добавил вывод реальных идентификаторов (UID и GID) и повторно компилируйте: (рис. 2.6), (рис. 2.7) и (рис. 2.8)



```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.8 simpleid2.c Modifié  
#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. 2.6: Усложнение программы, создание simpleid2.c

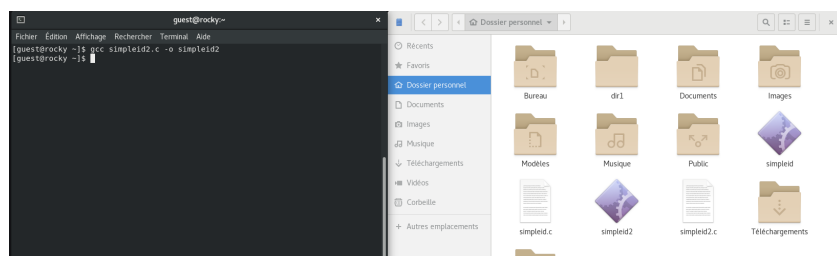


Рис. 2.7: Усложнение программы, создание simpleid2.c


```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ gcc simpleid2.c -o simpleid2  
[guest@rocky ~]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@rocky ~]$
```

Рис. 2.8: Усложнение программы, создание simpleid2.c

5. Изменение владельца программы simpleid2 на root и установка SetUID-бита:

- Использовал следующие команды: (рис. 2.9)

```
root@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ su -  
Mot de passe :  
[root@rocky ~]# chown root:guest /home/guest/simpleid2  
[root@rocky ~]# chmod u+s /home/guest/simpleid2  
[root@rocky ~]#
```

Рис. 2.9: Изменение владельца программы

- Использовал sudo или повысьте временно свои права с помощью su (рис. 2.10).

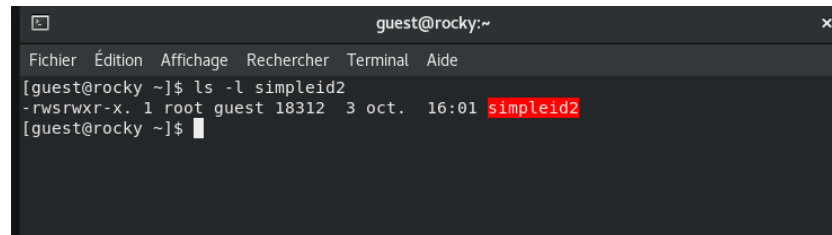
```
guest@rocky:/home/guest  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ sudo  
usage: sudo -h | -K | -k | -V  
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]  
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user]  
[command]  
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-D directory] [-g group]  
[-h host] [-p prompt] [-R directory] [-T timeout] [-u user]  
[VAR=value] [-i|-s] [<command>]  
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-D directory] [-g group]  
[-h host] [-p prompt] [-R directory] [-T timeout] [-u user] file ...  
[guest@rocky ~]$ su  
Mot de passe :  
[root@rocky guest]#
```

Рис. 2.10: Изменение владельца программы

- Это позволит программе выполняться с правами пользователя root, независимо от того, кто её запускает.

6. Запуск `simpleid2` и `id` для сравнения результатов:

- Сначала выполнил проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`. (рис. 2.11)



```

guest@rocky:~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[guest@rocky ~]$ ls -l simpleid2
-rwsrwxr-x. 1 root guest 18312  3 oct.  16:01 simpleid2
[guest@rocky ~]$

```

Рис. 2.11: Запуск `simpleid2` и `id` для сравнения результатов

- Потом запустил программу и убедился, что эффективный UID соответствует root, даже если программа запускается обычным пользователем, что подтверждает работу SetUID (рис. 2.12).



```

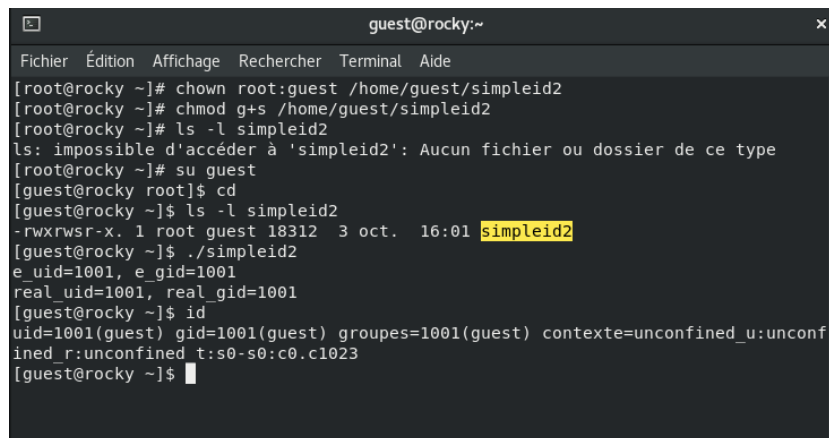
guest@rocky:~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
[guest@rocky ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@rocky ~]$ id
uid=1001(guest) gid=1001(guest) groupes=1001(guest) contexte=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@rocky ~]$

```

Рис. 2.12: Запуск `simpleid2` и `id` для сравнения результатов

7. Применение SetGID-бита к `simpleid2`:

- Изменил группу файла и установил SetGID: (рис. 2.13)



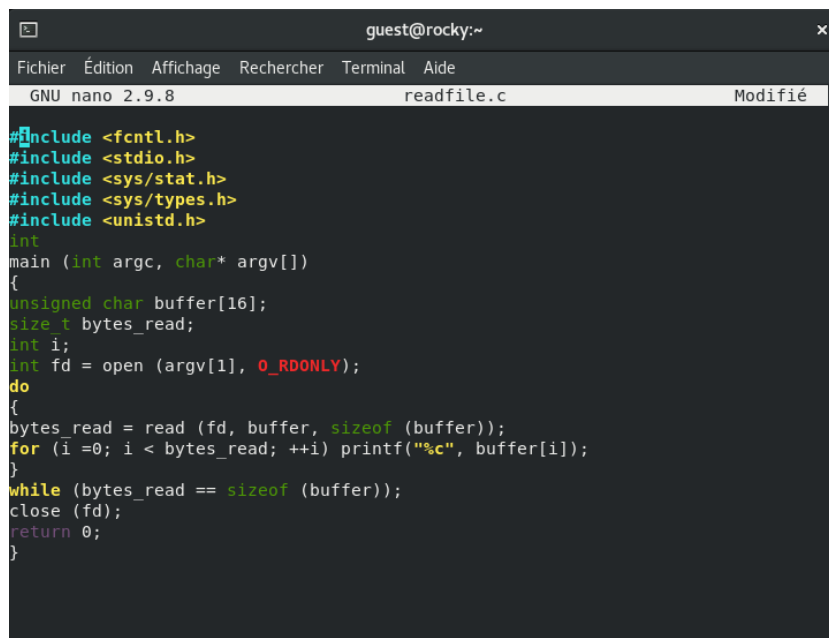
```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[root@rocky ~]# chown root:guest /home/guest/simpleid2  
[root@rocky ~]# chmod g+s /home/guest/simpleid2  
[root@rocky ~]# ls -l simpleid2  
ls: impossible d'accéder à 'simpleid2': Aucun fichier ou dossier de ce type  
[root@rocky ~]# su guest  
[guest@rocky root]$ cd  
[guest@rocky ~]$ ls -l simpleid2  
-rwxrwsr-x. 1 root guest 18312 3 oct. 16:01 simpleid2  
[guest@rocky ~]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@rocky ~]$ id  
uid=1001(guest) gid=1001(guest) groupes=1001(guest) contexte=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@rocky ~]$
```

Рис. 2.13: Применение SetGID-бита к simpleid2

- Программа будет выполняться с правами группы guest. Сравним результаты `./simpleid2` и `id`, чтобы увидеть, что GID соответствует группе владельца, указанной в SetGID.

8. Создание программы `readfile.c` для чтения файла:

- Создал программу `readfile.c`, которая читает содержимое файла, и скомпилируйте её: (рис. 2.14) и (рис. 2.15)



```
GNU nano 2.9.8 readfile.c  
#include <fcntl.h>  
#include <stdio.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
#include <unistd.h>  
int  
main (int argc, char* argv[])  
{  
    unsigned char buffer[16];  
    size_t bytes_read;  
    int i;  
    int fd = open (argv[1], O_RDONLY);  
    do  
    {  
        bytes_read = read (fd, buffer, sizeof (buffer));  
        for (i = 0; i < bytes_read; ++i) printf("%c", buffer[i]);  
    }  
    while (bytes_read == sizeof (buffer));  
    close (fd);  
    return 0;  
}
```

Рис. 2.14: Создание программы `readfile.c` для чтения файла

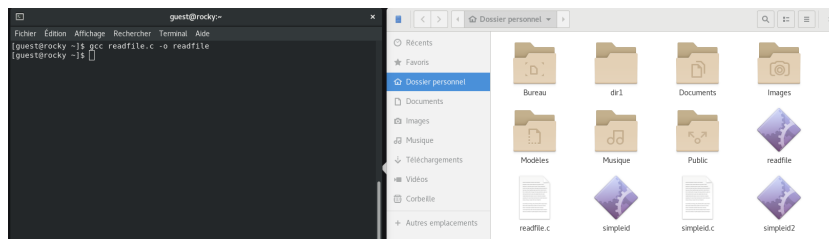


Рис. 2.15: Создание программы readfile.c для чтения файла

9. Изменение владельца файла и прав доступа:

- Сменил владельца файла readfile.c на root, установил права так, чтобы только root мог его читать: (рис. 2.16)

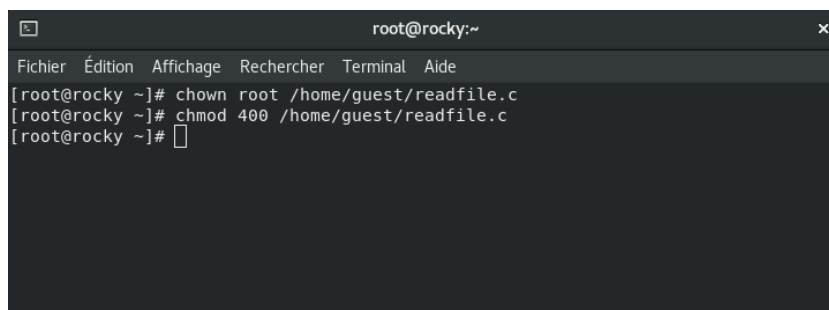


Рис. 2.16: Изменение владельца файла и прав доступа

- Проверил, что пользователь guest не может прочитать файл readfile.c (должна быть ошибка “Permission denied”) (рис. 2.17).

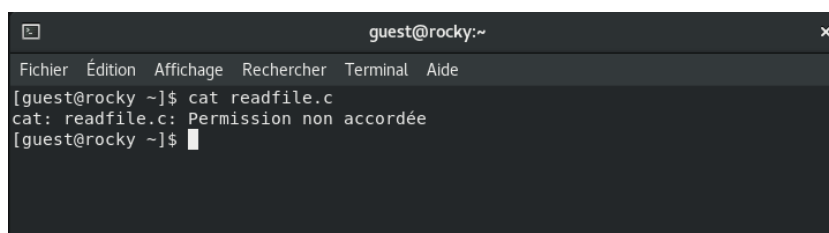
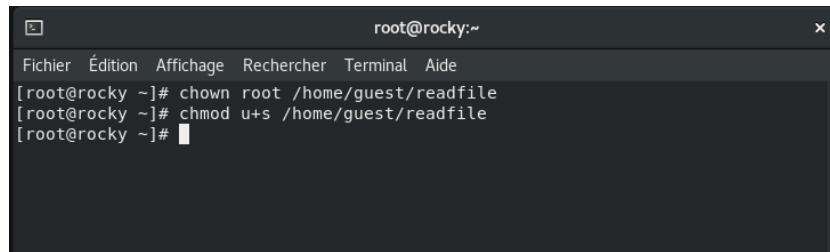


Рис. 2.17: Проверка прочтения

10. Установка SetUID-бита на программу readfile:

- Сменил владельца на root и установите SetUID: (рис. 2.18)

A terminal window titled 'root@rocky:~' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Terminal', and 'Aide'. The terminal shows the following commands and output:

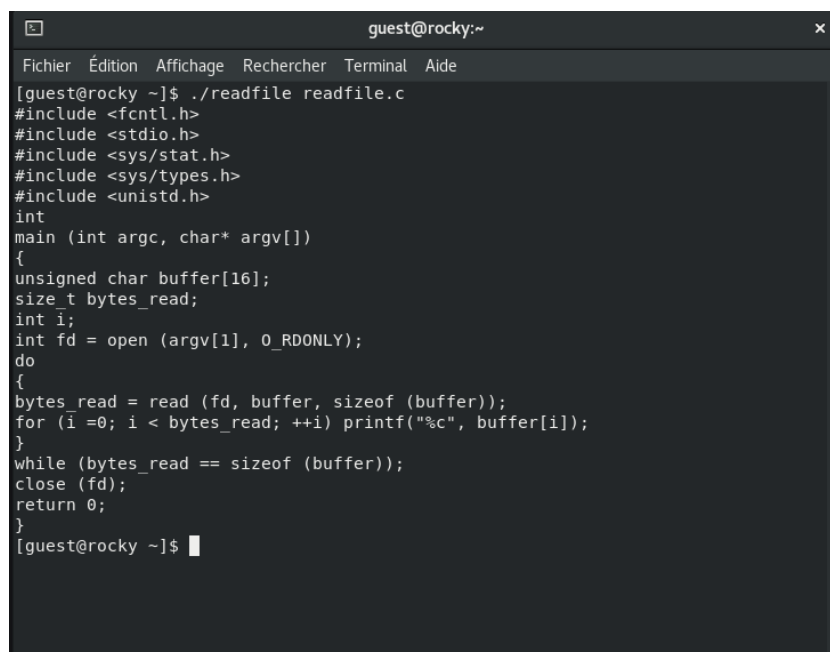
```
[root@rocky ~]# chown root /home/guest/readfile
[root@rocky ~]# chmod u+s /home/guest/readfile
[root@rocky ~]#
```

Рис. 2.18: Установка SetUID-бита на программу readfile

- Теперь программа будет выполняться с правами root.

11. Проверка, может ли программа readfile прочитать файл readfile.c:

- Запустите readfile как guest: (рис. 2.19)

A terminal window titled 'guest@rocky:~' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Terminal', and 'Aide'. The terminal shows the execution of a program and the source code of readfile.c:

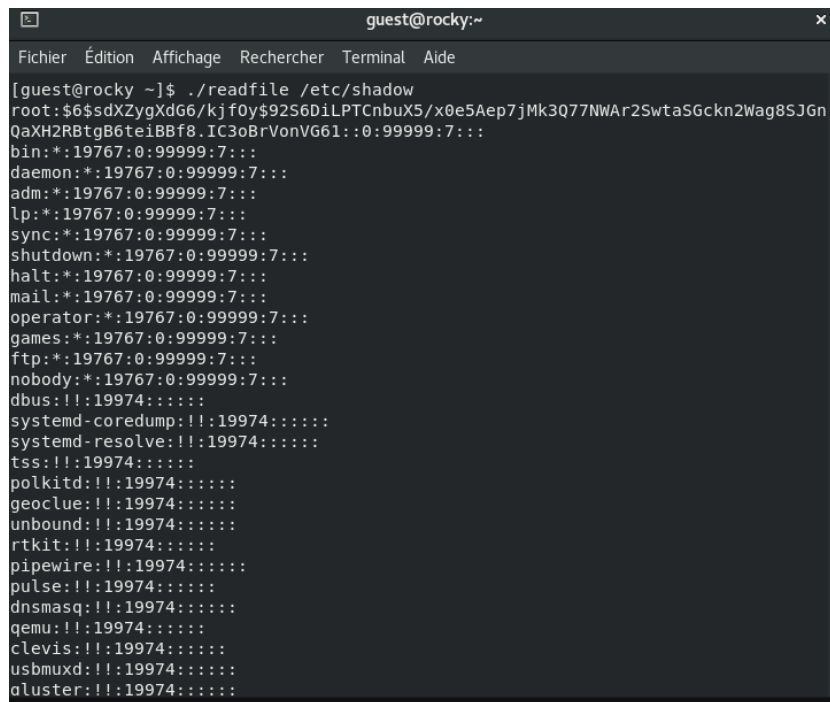
```
[guest@rocky ~]$ ./readfile readfile.c
#include <fcntl.h>
#include <stdio.h>
#include <sys/stat.h>
#include <sys/types.h>
#include <unistd.h>
int
main (int argc, char* argv[])
{
    unsigned char buffer[16];
    size_t bytes_read;
    int i;
    int fd = open (argv[1], O_RDONLY);
    do
    {
        bytes_read = read (fd, buffer, sizeof (buffer));
        for (i=0; i < bytes_read; ++i) printf("%c", buffer[i]);
    }
    while (bytes_read == sizeof (buffer));
    close (fd);
    return 0;
}
[guest@rocky ~]$
```

Рис. 2.19: Проверка, может ли программа readfile прочитать файл readfile.c

- Программа должна успешно прочитать файл благодаря SetUID-биту.

12. Проверка, может ли программа readfile прочитать файл /etc/shadow:

- Запустил ./readfile /etc/shadow. Если SetUID-бит установлен, программа сможет читать этот файл, что демонстрирует возможность использования SetUID для доступа к привилегированным данным (рис. 2.20).



```
guest@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest@rocky ~]$ ./readfile /etc/shadow  
root:$6$sdXZygXdG6/kjf0y$92S6DiLPTCnbuX5/x0e5Aep7jMk3Q77NWA2SwtasGckn2Wag8SJGn  
QaXH2RbtgB6te1BBf8.IC3oBrVonVG61::0:99999:7:::  
bin:!:19767:0:99999:7:::  
daemon:!:19767:0:99999:7:::  
adm:!:19767:0:99999:7:::  
lp:!:19767:0:99999:7:::  
sync:!:19767:0:99999:7:::  
shutdown:!:19767:0:99999:7:::  
halt:!:19767:0:99999:7:::  
mail:!:19767:0:99999:7:::  
operator:!:19767:0:99999:7:::  
games:!:19767:0:99999:7:::  
ftp:!:19767:0:99999:7:::  
nobody:!:19767:0:99999:7:::  
dbus:!!:19974:::  
systemd-coredump:!!:19974:::  
systemd-resolve:!!:19974:::  
tss:!!:19974:::  
polkitd:!!:19974:::  
geoclue:!!:19974:::  
unbound:!!:19974:::  
rtkit:!!:19974:::  
pipewire:!!:19974:::  
pulse:!!:19974:::  
dnsmasq:!!:19974:::  
qemu:!!:19974:::  
clevis:!!:19974:::  
usbmuxd:!!:19974:::  
cluster:!!:19974:::
```

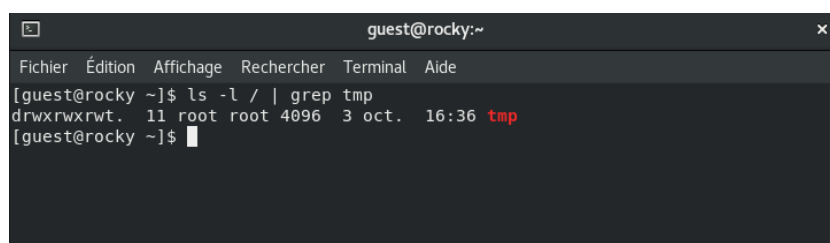
Рис. 2.20: Проверка, может ли программа readfile прочитать файл /etc/shadow

2.2 Часть 2: Исследование Sticky-бита

2.2.1 Вопросы и Решение

1. Проверка наличия Sticky-бита на директории /tmp:

- Выполнил команду: (рис. 2.21)



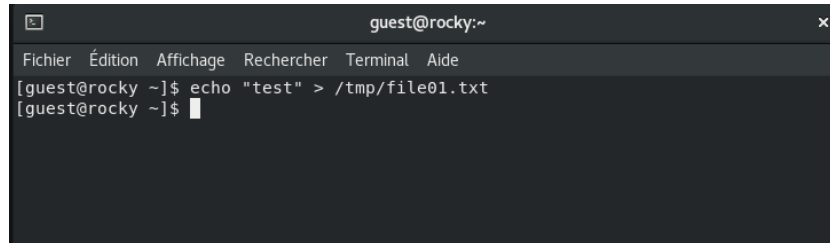
```
guest@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest@rocky ~]$ ls -l / | grep tmp  
drwxrwxrwt.  11 root root 4096  3 oct.  16:36 tmp  
[guest@rocky ~]$
```

Рис. 2.21: Проверка наличия Sticky-бита на директории /tmp

- Sticky-бит (t) указывает, что только владелец файла или root может удалить его, даже если у других пользователей есть права на запись.

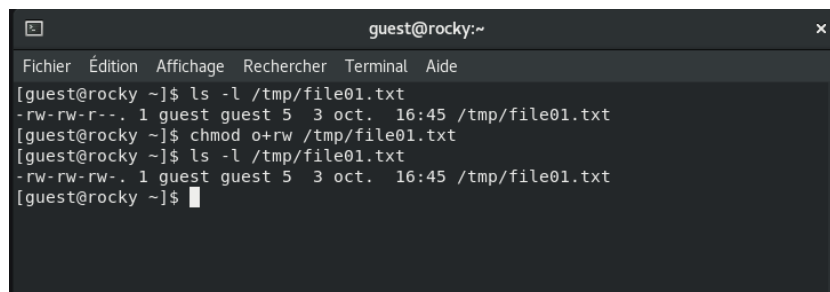
2. Создание файла в /tmp и изменение прав доступа:

- Как пользователь guest, создайте файл file01.txt в /tmp и разрешил записать всем: (рис. 2.22) и (рис. 2.23)



```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ echo "test" > /tmp/file01.txt  
[guest@rocky ~]$
```

Рис. 2.22: Создание файла в /tmp и изменение прав доступа

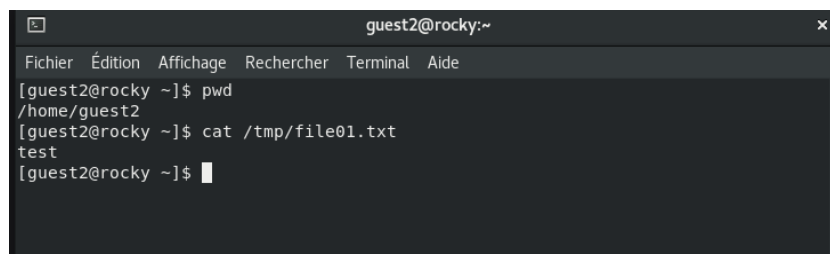


```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ ls -l /tmp/file01.txt  
-rw-rw-r--. 1 guest guest 5 3 oct. 16:45 /tmp/file01.txt  
[guest@rocky ~]$ chmod o+rw /tmp/file01.txt  
[guest@rocky ~]$ ls -l /tmp/file01.txt  
-rw-rw-rw-. 1 guest guest 5 3 oct. 16:45 /tmp/file01.txt  
[guest@rocky ~]$
```

Рис. 2.23: Создание файла в /tmp и изменение прав доступа

3. Чтение и изменение файла другим пользователем guest2:

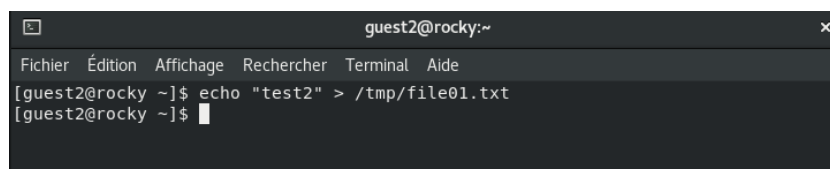
- Как пользователь guest2, попытался: (рис. 2.24)



```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ pwd  
/home/guest2  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test  
[guest2@rocky ~]$
```

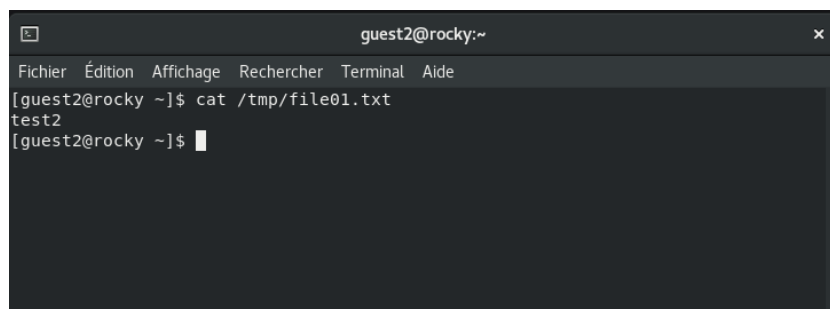
Рис. 2.24: Чтение и изменение файла другим пользователем guest2

- Чтение и запись должны быть успешными, но удаление файла будет невозможно благодаря Sticky-биту (рис. 2.25), (рис. 2.26), (рис. 2.27), (рис. 2.28) и (рис. 2.29).



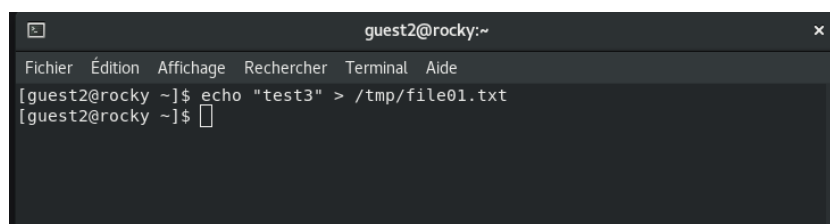
```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ echo "test2" > /tmp/file01.txt  
[guest2@rocky ~]$
```

Рис. 2.25: Чтение и изменение файла другим пользователем guest2



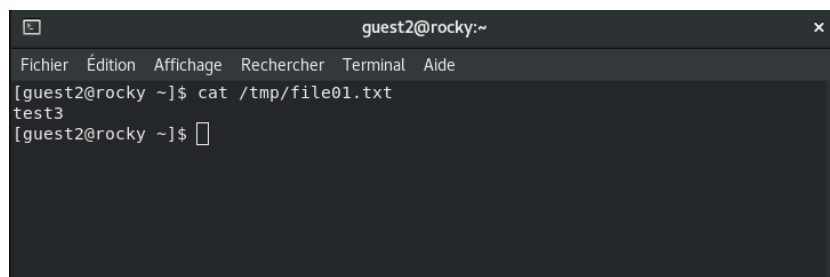
```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test2  
[guest2@rocky ~]$
```

Рис. 2.26: Чтение и изменение файла другим пользователем guest2



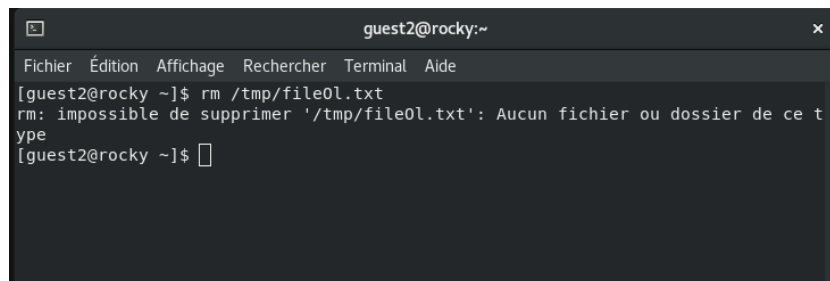
```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ echo "test3" > /tmp/file01.txt  
[guest2@rocky ~]$
```

Рис. 2.27: Чтение и изменение файла другим пользователем guest2



```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test3  
[guest2@rocky ~]$
```

Рис. 2.28: Чтение и изменение файла другим пользователем guest2

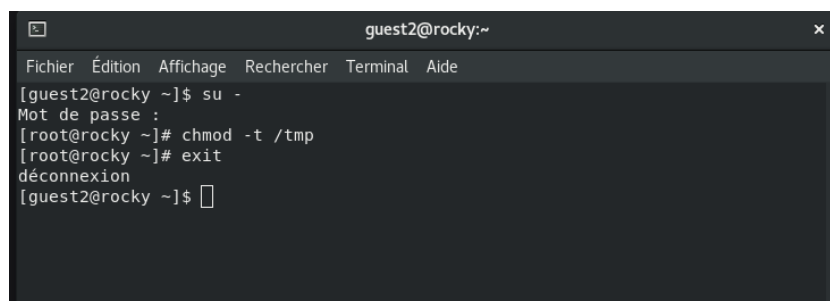


```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ rm /tmp/file0l.txt  
rm: impossible de supprimer '/tmp/file0l.txt': Aucun fichier ou dossier de ce t  
ype  
[guest2@rocky ~]$
```

Рис. 2.29: Чтение и изменение файла другим пользователем guest2

4. Удаление Sticky-бита с /tmp:

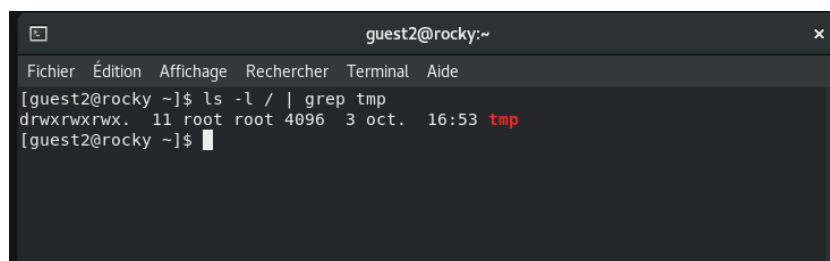
- Как суперпользователь, снимите Sticky-бит: (рис. 2.30)



```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ su -  
Mot de passe :  
[root@rocky ~]# chmod -t /tmp  
[root@rocky ~]# exit  
déconnexion  
[guest2@rocky ~]$
```

Рис. 2.30: Удаление Sticky-бита с /tmp

- От пользователя guest2 проверил, что атрибута t у директории /tmp нет: (рис. 2.31)



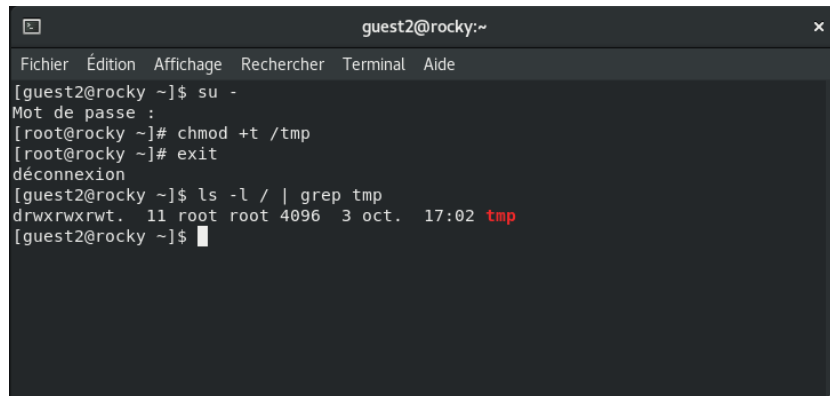
```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ ls -l / | grep tmp  
drwxrwxrwx. 11 root root 4096 3 oct. 16:53 tmp  
[guest2@rocky ~]$
```

Рис. 2.31: Проверка атрибута

- Повторил попытку удаления файла от пользователя guest2. Теперь файл должен быть удалён, так как Sticky-бит больше не защищает его, но что-то не так.

5. Возвращение Sticky-бита на /tmp:

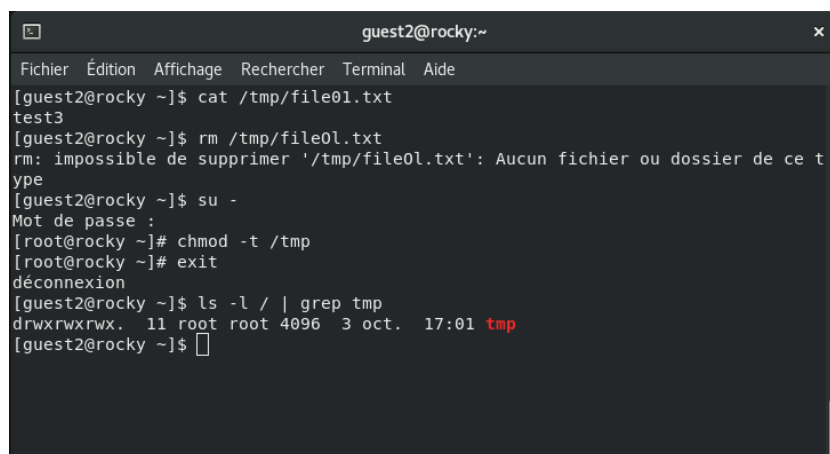
- Повторно установил Sticky-бит: (рис. 2.32)



```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ su -  
Mot de passe :  
[root@rocky ~]# chmod +t /tmp  
[root@rocky ~]# exit  
déconnexion  
[guest2@rocky ~]$ ls -l / | grep tmp  
drwxrwxrwt. 11 root root 4096 3 oct. 17:02 tmp  
[guest2@rocky ~]$
```

Рис. 2.32: Возвращение Sticky-бита на /tmp

- Теперь повторил действия и убедился, что удаление файла снова запрещено для пользователей, не являющихся его владельцем (рис. 2.33).



```
guest2@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test3  
[guest2@rocky ~]$ rm /tmp/file01.txt  
rm: impossible de supprimer '/tmp/file01.txt': Aucun fichier ou dossier de ce t  
ype  
[guest2@rocky ~]$ su -  
Mot de passe :  
[root@rocky ~]# chmod -t /tmp  
[root@rocky ~]# exit  
déconnexion  
[guest2@rocky ~]$ ls -l / | grep tmp  
drwxrwxrwx. 11 root root 4096 3 oct. 17:01 tmp  
[guest2@rocky ~]$
```

Рис. 2.33: Повторение действия

3 Выводы

В ходе этой лабораторной работы было продемонстрировано, как использование **SetUID**, **SetGID**, и **Sticky-бита** позволяет управлять доступом и правами пользователей в системе Linux. **SetUID** и **SetGID** позволяют временно выполнять программы с правами владельца файла или его группы, что полезно для выполнения привилегированных задач. **Sticky-бит** защищает файлы в общих директориях, таких как `/tmp`, от удаления пользователями, не являющимися владельцами, что предотвращает потенциальные конфликты и нарушения безопасности.