

# Презентация по лабораторной работе №5

Информационная безопасность

---

Акандзо Жордани Лади Гаэл.

03 Октября 2024

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Акондзо Жордани Лади Гаэл.
- студент 4-го курса группы НКНбд-01-21
- 1032215649
- Российский университет дружбы народов
- GitHub

## Вводная часть

---

- Обеспечение безопасности
- Предотвращение пересечений между пользовательскими аккаунтами
- Совместный доступ к файлам

- Закрепление практических навыков работы в консоли с атрибутами файлов
- Закрепление теоретических основ дискреционного разграничения доступа в современных системах с открытым кодом на базе ОС Linux
- Изучение механизмов изменения идентификаторов, применение SetUID- и SetGID-битов.
- Изучение действия Sticky-бита на запись и удаление файлов в общей директории.

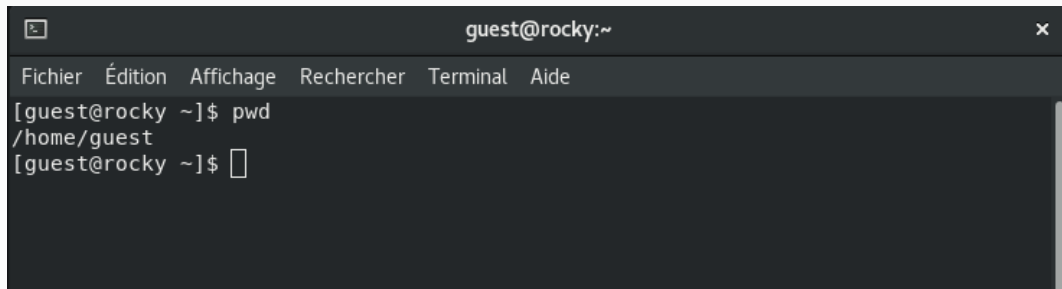
- Веб-сервис **GitHub** для работы с репозиториями
- Программа для виртуализации ОС **VirtualBox**
- Процессор **pandoc** для входного формата Markdown
- Результирующие форматы
  - pdf
  - docx
- Автоматизация процесса создания: **Makefile**

## Выполнение лабораторной работы

---



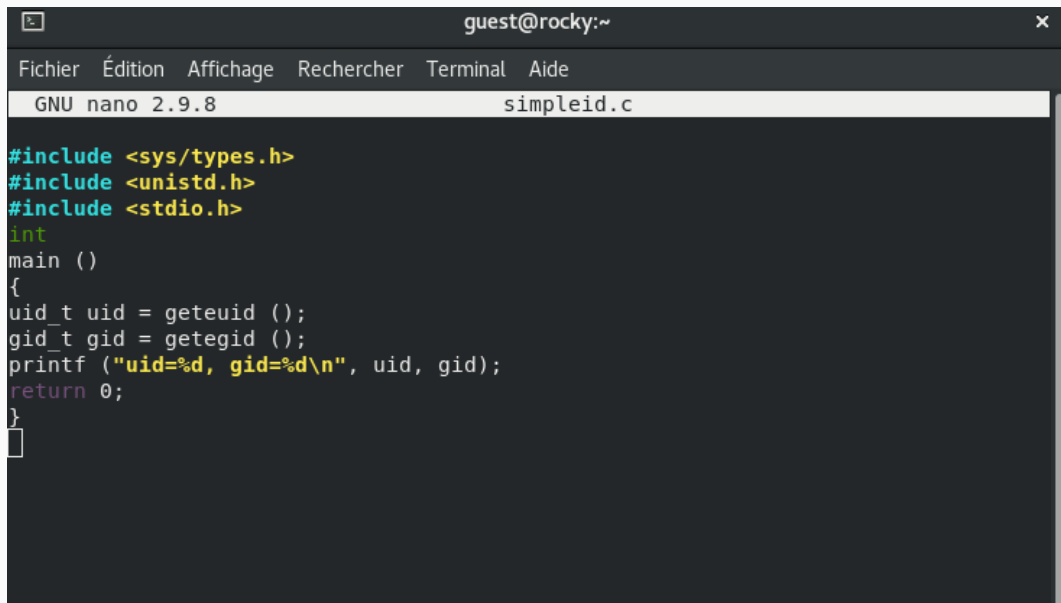




A terminal window titled "guest@rocky:~" with a standard Linux menu bar (Fichier, Édition, Affichage, Rechercher, Terminal, Aide). The terminal shows the user typing "pwd" and receiving the output "/home/guest".

```
guest@rocky:~  
[guest@rocky ~]$ pwd  
/home/guest  
[guest@rocky ~]$
```

## Создание программы simpleid.c и её компиляция:



The screenshot shows a terminal window titled "guest@rocky:~" with a menu bar containing "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". Below the menu bar, the text "GNU nano 2.9.8" and "simpleid.c" are displayed. The main area of the window contains the following C code:

```
#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;
}
```

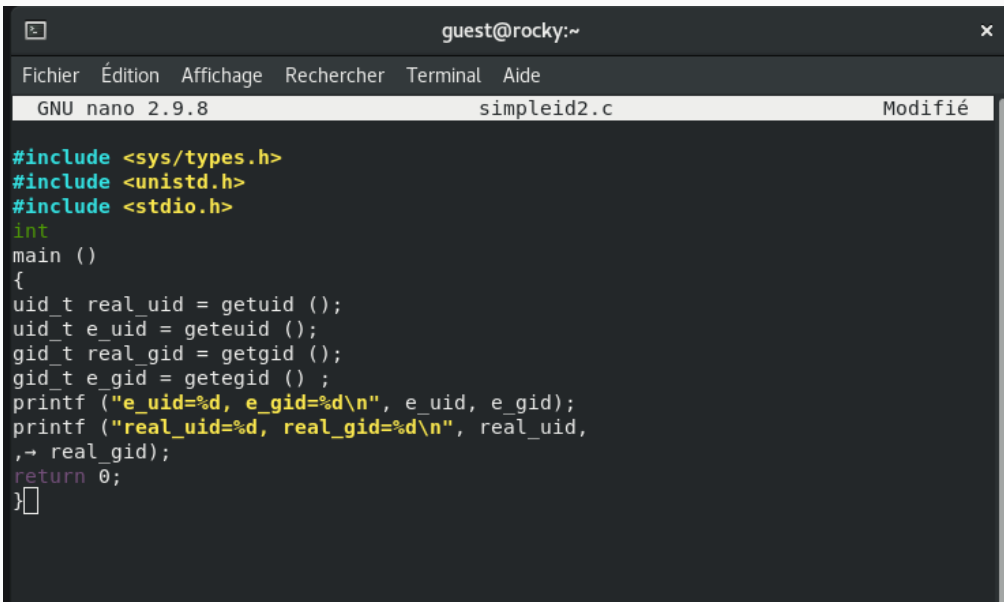
A cursor is visible at the end of the closing brace of the main function.

## Запуск программы `simpleid` и сравнение результата с системной командой `id`:

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

```
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 ~]$
```

## Усложнение программы, создание simpleid2.c:



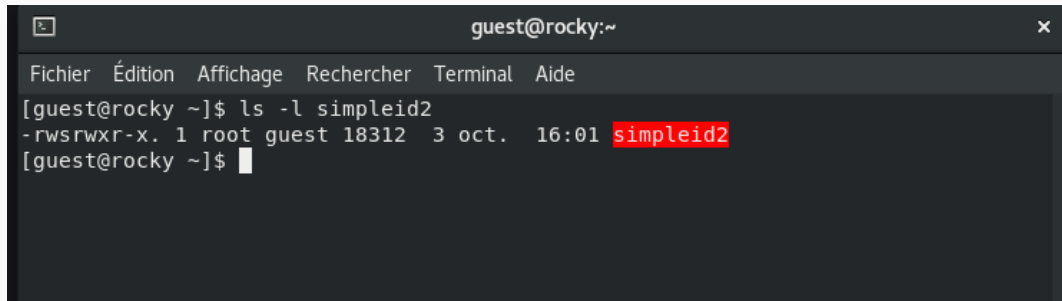
```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
GNU nano 2.9.8 simpleid2.c Modifié  
  
#include <sys/types.h>  
#include <unistd.h>  
#include <stdio.h>  
int  
main ()  
{  
    uid_t real_uid = getuid ();  
    uid_t e_uid = geteuid ();  
    gid_t real_gid = getgid ();  
    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;  
}
```

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

```
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 ~]#
```

```
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]
```

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



A terminal window titled "guest@rocky:~" with a menu bar containing "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". The terminal shows the command `[guest@rocky ~]$ ls -l simpleid2` and its output: `-rwsrwxr-x. 1 root guest 18312 3 oct. 16:01 simpleid2`. The filename `simpleid2` in the output is highlighted in red. The prompt `[guest@rocky ~]$` is followed by a cursor.

```
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 ~]$
```

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



```
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:unconf  
ined_r:unconfined t:s0-s0:c0.c1023  
[guest@rocky ~]$
```



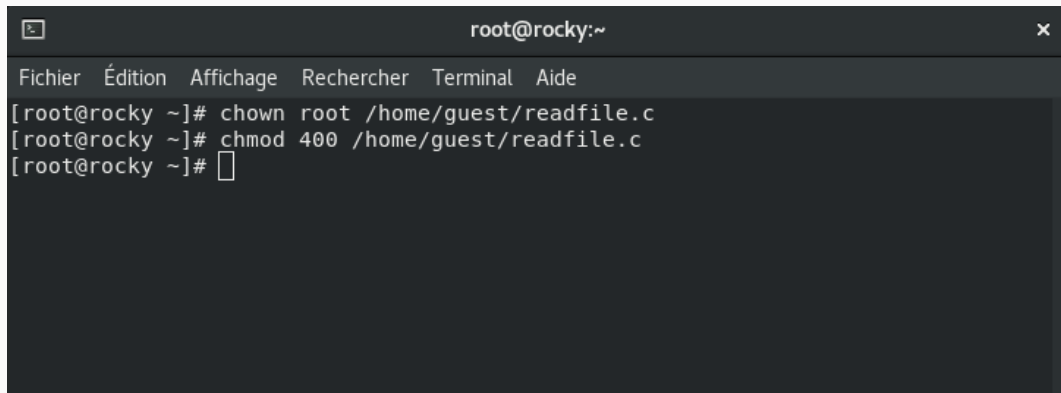
## Применение SetGID-бита к simpleid2:

```
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:unconf  
ined_r:unconfined t:s0-s0:c0.c1023  
[guest@rocky ~]$
```

## Создание программы readfile.c для чтения файла:

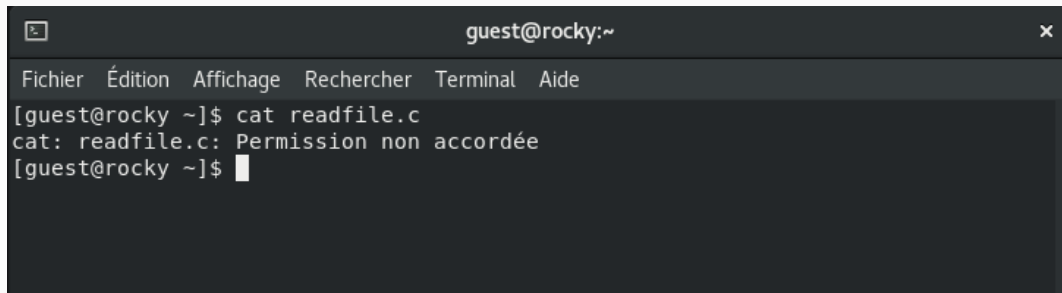
```
guest@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
GNU nano 2.9.8      readfile.c      Modifié  
  
#include <fcntl.h>  
#include <stdio.h>  
#include <sys/stat.h>  
#include <sys/types.h>  
#include <unistd.h>  
int  
main (int argc, char* argv[])  
{  
    unsigned char buffer[16];  
    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;  
}
```

## Изменение владельца файла и прав доступа:

A terminal window titled 'root@rocky:~' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Terminal', and 'Aide'. The terminal shows three commands being executed in sequence: 'chown root /home/guest/readfile.c', 'chmod 400 /home/guest/readfile.c', and a blank prompt line.

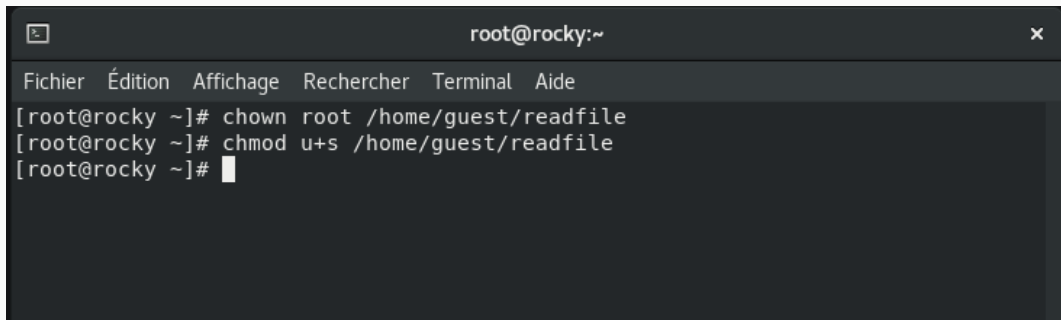
```
root@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[root@rocky ~]# chown root /home/guest/readfile.c  
[root@rocky ~]# chmod 400 /home/guest/readfile.c  
[root@rocky ~]#
```

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

A terminal window titled "guest@rocky:~" with a standard Linux menu bar (Fichier, Édition, Affichage, Rechercher, Terminal, Aide). The terminal shows the command "cat readfile.c" being executed, which results in the error message "cat: readfile.c: Permission non accordée". The prompt returns to the user.

```
guest@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest@rocky ~]$ cat readfile.c  
cat: readfile.c: Permission non accordée  
[guest@rocky ~]$
```

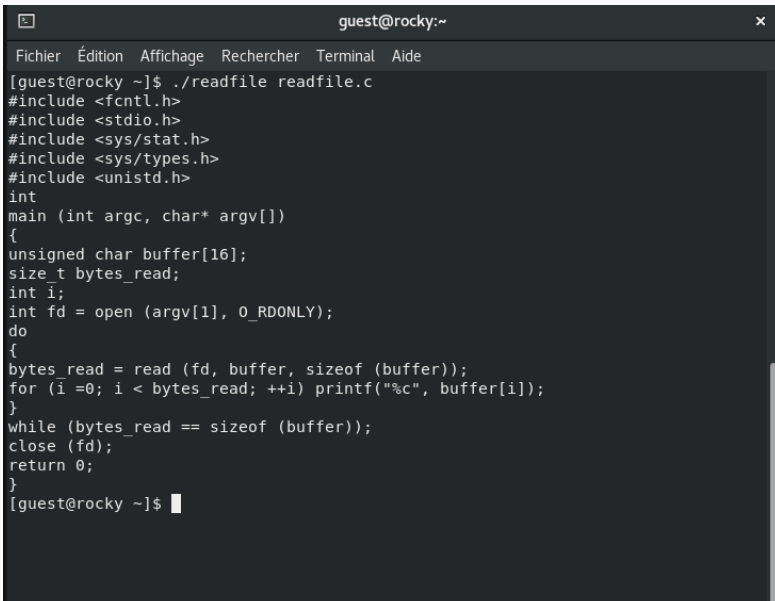
## Установка SetUID-бита на программу readfile:



```
root@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[root@rocky ~]# chown root /home/guest/readfile  
[root@rocky ~]# chmod u+s /home/guest/readfile  
[root@rocky ~]#
```

- Теперь программа будет выполняться с правами root.

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



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

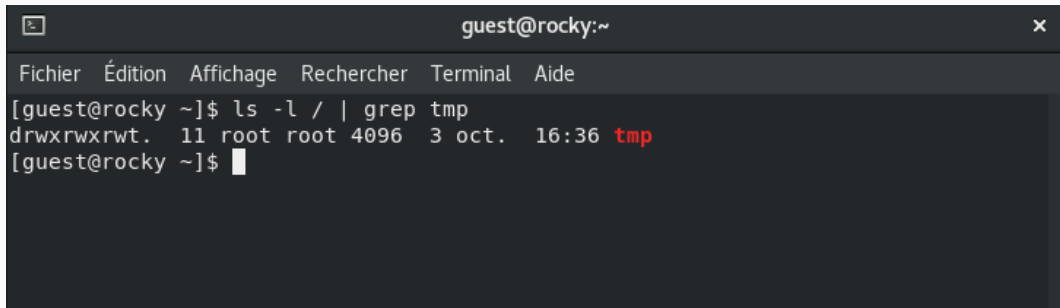
## Проверка, может ли программа readfile прочитать файл /etc/shadow:

```
guest@rocky:~  
Fichier Édition Affichage Rechercher Terminal Aide  
[guest@rocky ~]$ ./readfile /etc/shadow  
root:$6$sdXZygXdG6/kjf0y$92S6DiLPTCnbuX5/x0e5Aep7jMk3Q77NwAr2SwtaSGckn2Wag8SJGn  
QaXH2RBtgB6teiBBf8.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:::~:  
aluster:!!:19974:::~:
```





## Проверка наличия Sticky-бита на директории /tmp:



```
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 ~]$
```

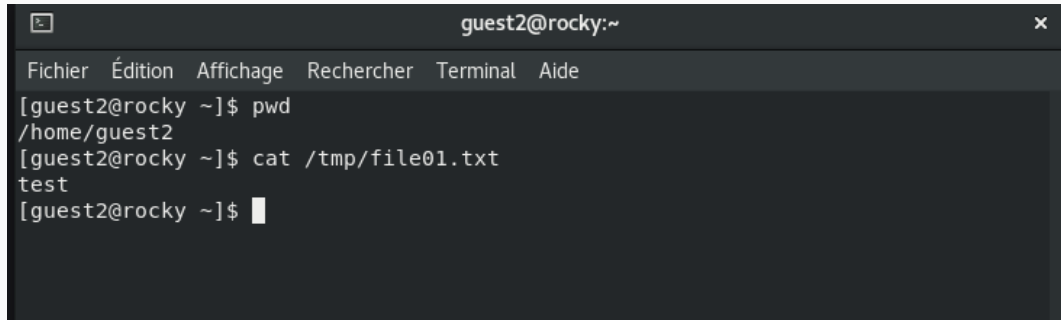
- Sticky-бит (**t**) указывает, что только владелец файла или root может удалить его, даже если у других пользователей есть права на запись.

## Создание файла в /tmp и изменение прав доступа:\*\*

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

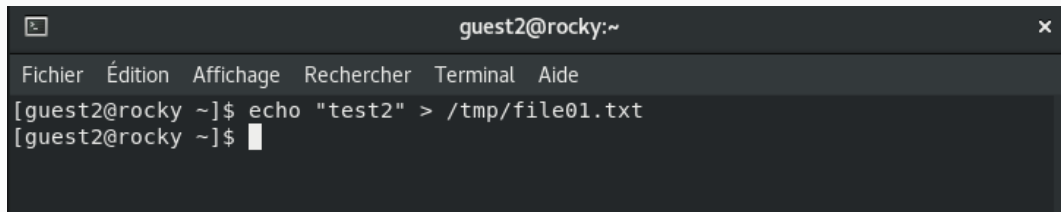
```
guest@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest@rocky ~]$ ls -l /tmp/file01.txt  
-rw-rw-r--. 1 guest guest 5  3 oct.  16:45 /tmp/file01.txt  
[guest@rocky ~]$ chmod o+rw /tmp/file01.txt  
[guest@rocky ~]$ ls -l /tmp/file01.txt  
-rw-rw-rw-. 1 guest guest 5  3 oct.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

## Чтение и изменение файла другим пользователем guest2:



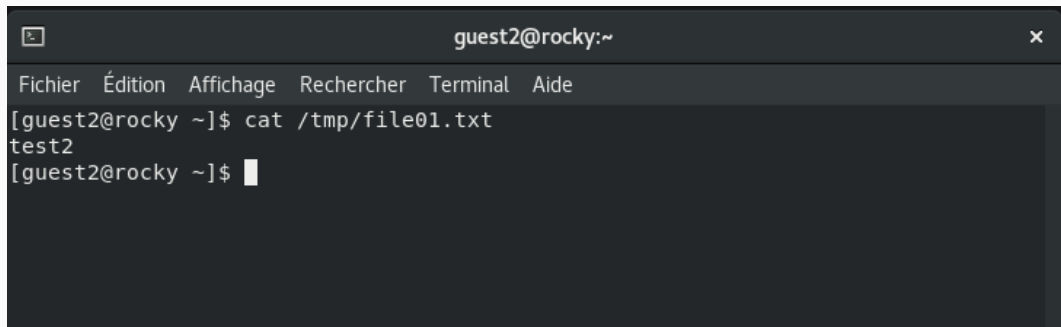
```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ pwd  
/home/guest2  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test  
[guest2@rocky ~]$
```

Чтение и запись должны быть успешными, но удаление файла будет невозможно благодаря Sticky-биту



A terminal window titled "guest2@rocky:~" with a menu bar containing "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". The terminal shows the command `echo "test2" > /tmp/file01.txt` being executed successfully, followed by a new prompt line.

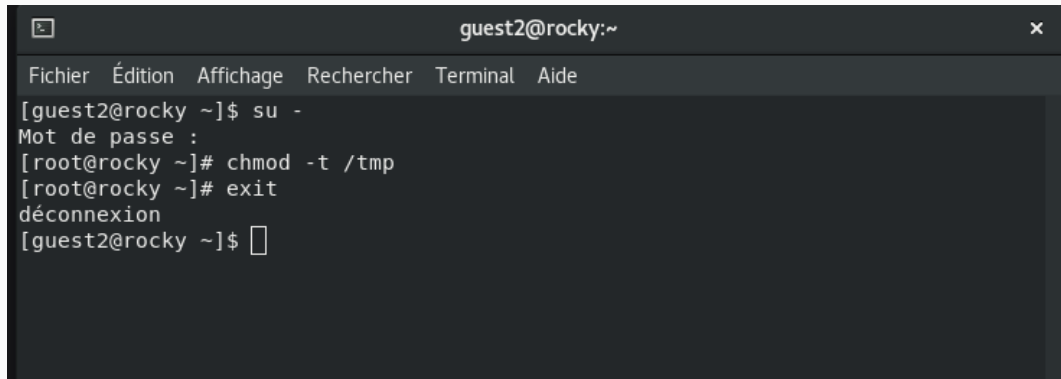
```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ echo "test2" > /tmp/file01.txt  
[guest2@rocky ~]$
```



A terminal window titled "guest2@rocky:~" with a menu bar containing "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". The terminal shows the command `cat /tmp/file01.txt` being executed, displaying the output "test2", followed by a new prompt line.

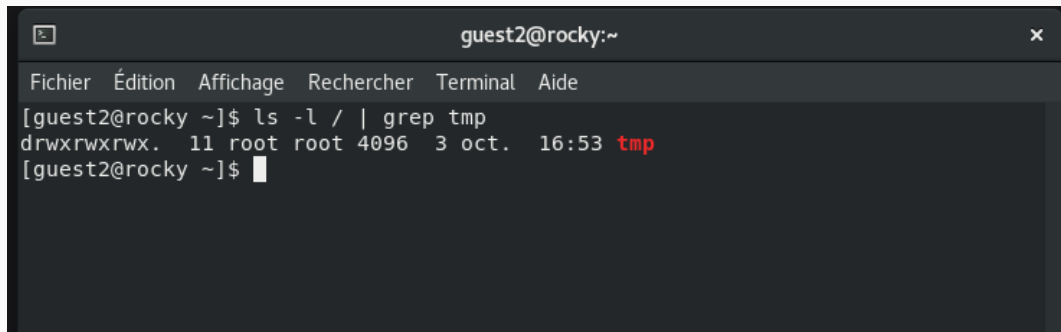
```
guest2@rocky:~  
Fichier  Édition  Affichage  Rechercher  Terminal  Aide  
[guest2@rocky ~]$ cat /tmp/file01.txt  
test2  
[guest2@rocky ~]$
```

## Удаление Sticky-бита с /tmp:

A terminal window titled 'guest2@rocky:~' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Terminal', and 'Aide'. The terminal shows a user switching to root and removing the sticky bit from /tmp.

```
guest2@rocky ~]$ su -  
Mot de passe :  
[root@rocky ~]# chmod -t /tmp  
[root@rocky ~]# exit  
déconnexion  
[guest2@rocky ~]$
```

От пользователя guest2 проверил, что атрибута t у директории /tmp нет:



A terminal window titled "guest2@rocky:~" with a menu bar containing "Fichier", "Édition", "Affichage", "Rechercher", "Terminal", and "Aide". The terminal shows the command `[guest2@rocky ~]$ ls -l / | grep tmp` and its output: `drwxrwxrwx. 11 root root 4096 3 oct. 16:53 tmp`. The word "tmp" in the output is highlighted in red. The prompt `[guest2@rocky ~]$` is followed by a cursor.

```
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 ~]$
```

- 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.

## Возвращение Sticky-бита на /tmp:



```
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 ~]$
```

Теперь повторил действия и убедился, что удаление файла снова запрещено для пользователей, не являющихся его владельцем.

```
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 ~]$
```



## 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.