

Дискреционное разграничение прав в Linux. Расширенные атрибуты

Бансимба Клодели Дъегра НПИбд-02-22¹

13 апреля 2024, Москва, Россия

¹Российский Университет Дружбы Народов

Информация

- Бансимба Клодели Дьегра
- Студент, НПИбд-02-22
- Российский университет дружбы народов
- 1032215651@pfur.ru



Цели и задачи работы

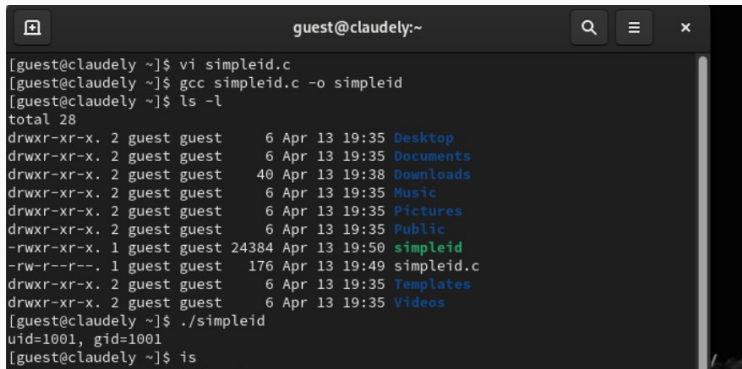
Цель лабораторной работы

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

Процесс выполнения лабораторной работы

Создание программы

Написали программу simpleid.c. Скомпилировали программу и убедились в

A terminal window titled 'guest@claudely:~' with search, menu, and close buttons. It shows the following commands and output:

```
[guest@claudely ~]$ vi simpleid.c
[guest@claudely ~]$ gcc simpleid.c -o simpleid
[guest@claudely ~]$ ls -l
total 28
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Desktop
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Documents
drwxr-xr-x. 2 guest guest  40 Apr 13 19:38 Downloads
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Music
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Pictures
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Public
-rwxr-xr-x. 1 guest guest 24384 Apr 13 19:50 simpleid
-rw-r--r--. 1 guest guest  176 Apr 13 19:49 simpleid.c
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Templates
drwxr-xr-x. 2 guest guest   6 Apr 13 19:35 Videos
[guest@claudely ~]$ ./simpleid
uid=1001, gid=1001
[guest@claudely ~]$ is
```

Рис. 1: программа simpleid

Создание программы

Усложнили программу, добавив вывод действительных идентификаторов.



```
guest@claudely:~ — /usr/bin/vim simpleid2.c
#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: программа simpleid2

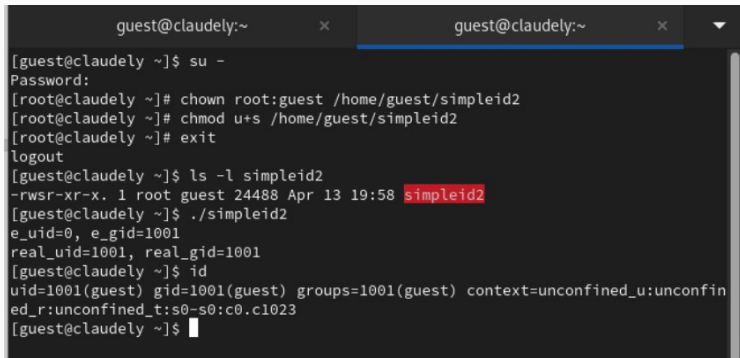
Скомпилировали и запустили simpleid2.c:



```
guest@claudely:~  
[guest@claudely ~]$ id  
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[guest@claudely ~]$ vi simpleid.c  
[guest@claudely ~]$ vi simpleid2.c  
[guest@claudely ~]$ gcc simpleid2.c -o simpleid2  
[guest@claudely ~]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@claudely ~]$
```

Рис. 3: запустили simpleid2

Запустили simpleid2 и id:

A terminal window with two tabs, both labeled 'guest@claudely:~'. The first tab shows the command 'su -' being executed, followed by a password prompt. The second tab shows the user as root, where the command 'chown root:guest /home/guest/simpleid2' is run, followed by 'chmod u+s /home/guest/simpleid2' and 'exit'. The first tab is active and shows the user back as guest. The command 'ls -l simpleid2' is run, showing the file permissions '-rwsr-xr-x. 1 root guest 24488 Apr 13 19:58 simpleid2', with 'simpleid2' highlighted in red. Then, './simpleid2' is run, displaying 'e_uid=0, e_gid=1001' and 'real_uid=1001, real_gid=1001'. Finally, 'id' is run, showing 'uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023'.

```
guest@claudely:~ x guest@claudely:~ x
[guest@claudely ~]$ su -
Password:
[root@claudely ~]# chown root:guest /home/guest/simpleid2
[root@claudely ~]# chmod u+s /home/guest/simpleid2
[root@claudely ~]# exit
logout
[guest@claudely ~]$ ls -l simpleid2
-rwsr-xr-x. 1 root guest 24488 Apr 13 19:58 simpleid2
[guest@claudely ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@claudely ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@claudely ~]$
```

Рис. 4: запустили simpleid2 и id

Откомпилировали её.

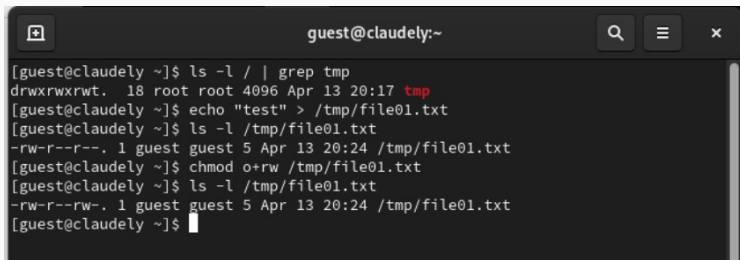
A terminal window with a dark background and light-colored text. It shows a sequence of four commands entered at the prompt [guest@claudely ~]\$. The first command is 'vi readfile.c', followed by an empty line. The third command is 'gcc readfile.c -o readfile', followed by an empty line. The fourth command is also an empty line.

```
[guest@claudely ~]$ vi readfile.c
[guest@claudely ~]$
[guest@claudely ~]$ gcc readfile.c -o readfile
[guest@claudely ~]$
```

Рис. 5: программа readfile

Исследование Sticky-бита

Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные»:

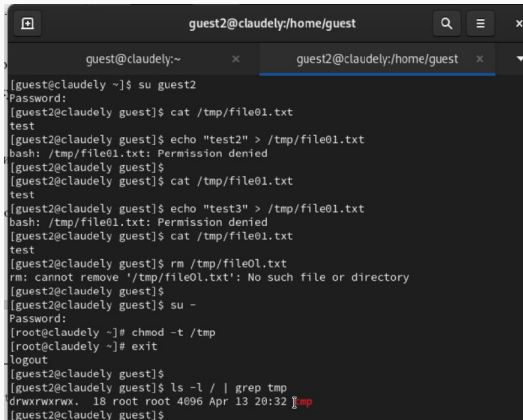


```
guest@claudely:~  
[guest@claudely ~]$ ls -l / | grep tmp  
drwxrwxrwt. 18 root root 4096 Apr 13 20:17 tmp  
[guest@claudely ~]$ echo "test" > /tmp/file01.txt  
[guest@claudely ~]$ ls -l /tmp/file01.txt  
-rw-r--r--. 1 guest guest 5 Apr 13 20:24 /tmp/file01.txt  
[guest@claudely ~]$ chmod o+rw /tmp/file01.txt  
[guest@claudely ~]$ ls -l /tmp/file01.txt  
-rw-r--rw-. 1 guest guest 5 Apr 13 20:24 /tmp/file01.txt  
[guest@claudely ~]$
```

Рис. 6: исследование Sticky-бита

Исследование Sticky-бита

От пользователя проверили, что атрибута `t` у директории `/tmp` нет:



```
guest2@claudely:/home/guest
[guest@claudely ~]$ su guest2
Password:
[guest2@claudely guest]$ cat /tmp/file01.txt
test
[guest2@claudely guest]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@claudely guest]$
[guest2@claudely guest]$ cat /tmp/file01.txt
test
[guest2@claudely guest]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Permission denied
[guest2@claudely guest]$ cat /tmp/file01.txt
test
[guest2@claudely guest]$ rm /tmp/file01.txt
rm: cannot remove '/tmp/file01.txt': No such file or directory
[guest2@claudely guest]$
[guest2@claudely guest]$ su -
Password:
[root@claudely ~]# chmod -t /tmp
[root@claudely ~]# exit
logout
[guest2@claudely guest]$
[guest2@claudely guest]$ ls -l / | grep tmp
drwxrwxrwx. 18 root root 4096 Apr 13 20:32 tmp
[guest2@claudely guest]$
```

Рис. 7: исследование Sticky-бита

Выводы по проделанной работе

Изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами.