

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Нгуен Тхай Зыонг НПИбд-01-19

6 октября, 2022, Москва, Россия

Российский Университет Дружбы Народов

Цели и задачи

- SUID - разрешение на установку идентификатора пользователя. Это бит разрешения, который позволяет пользователю запускать исполняемый файл с правами владельца этого файла.
- SGID - разрешение на установку идентификатора группы. Принцип работы очень похож на SUID с отличием, что файл будет запускаться пользователем от имени группы, которая владеет файлом.

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.

Выполнение лабораторной работы

Программа simpleid

```
[guest@nguenziong ~]$  
[guest@nguenziong ~]$ mkdir lab5  
[guest@nguenziong ~]$ touch simpleid.c  
[guest@nguenziong ~]$ touch simpleid2.c  
[guest@nguenziong ~]$ touch readfile.c  
[guest@nguenziong ~]$ gedit simpleid.c  
[guest@nguenziong ~]$ gcc simpleid.c  
[guest@nguenziong ~]$ gcc simpleid.c -o simpleid  
[guest@nguenziong ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@nguenziong ~]$ id  
uid=1001(guest) gid=1001(guest) rpyнны=1001(guest) контекст=unconfined_u:unconfi  
ned_r:unconfined_t:s0-s0:c0.c1023  
[guest@nguenziong ~]$
```

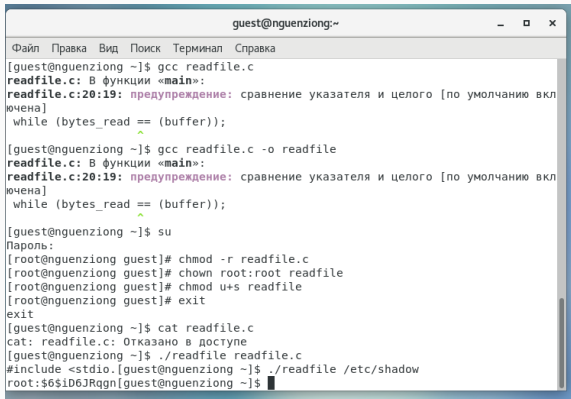
Figure 1: результат программы simpleid

Программа simpleid2

```
[guest@nguenziong ~]$  
[guest@nguenziong ~]$ gedit simpleid2.c  
[guest@nguenziong ~]$ gcc simpleid2.c  
[guest@nguenziong ~]$ gcc simpleid2.c -o simpleid2  
[guest@nguenziong ~]$ ./simpleid2  
e_uid=1001, e_gid=1001  
real_uid=1001, real_gid=1001  
[guest@nguenziong ~]$ su  
Пароль:  
[root@nguenziong guest]# chown root:guest simpleid2  
[root@nguenziong guest]# chmod u+s simpleid2  
[root@nguenziong guest]# ./simpleid2  
e_uid=0, e_gid=0  
real_uid=0, real_gid=0  
[root@nguenziong guest]# id  
uid=0(root) gid=0(root) rpyнны=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
[root@nguenziong guest]# chmod g+s simpleid2  
[root@nguenziong guest]# ./simpleid2  
e_uid=0, e_gid=1001  
real_uid=0, real_gid=0  
[root@nguenziong guest]# exit  
exit  
[guest@nguenziong ~]$ █
```

Figure 2: результат программы simpleid2

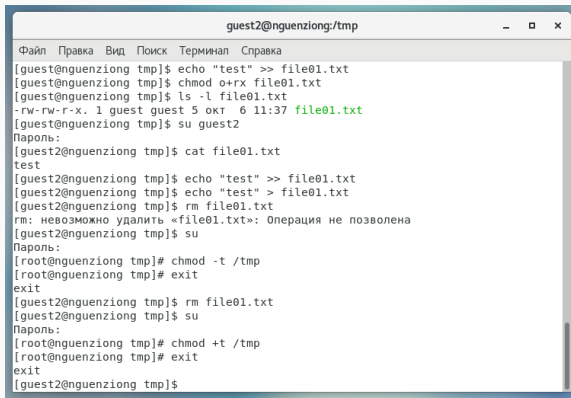
Программа readfile



```
guest@nguenziong:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
[guest@nguenziong ~]$ gcc readfile.c  
readfile.c: В функции «main»:  
readfile.c:20:19: предупреждение: сравнение указателя и целого [по умолчанию включена]  
    while (bytes_read == (buffer));  
                      ^  
[guest@nguenziong ~]$ gcc readfile.c -o readfile  
readfile.c: В функции «main»:  
readfile.c:20:19: предупреждение: сравнение указателя и целого [по умолчанию включена]  
    while (bytes_read == (buffer));  
                      ^  
[guest@nguenziong ~]$ su  
Пароль:  
[root@nguenziong guest]# chmod -r readfile.c  
[root@nguenziong guest]# chown root:root readfile  
[root@nguenziong guest]# chmod u+s readfile  
[root@nguenziong guest]# exit  
exit  
[guest@nguenziong ~]$ cat readfile.c  
cat: readfile.c: Отказано в доступе  
[guest@nguenziong ~]$ ./readfile readfile.c  
#include <stdio.h>[guest@nguenziong ~]$ ./readfile /etc/shadow  
root:$6$iD6JRqgn[guest@nguenziong ~]$
```

Figure 3: результат программы readfile

Исследование Sticky-бита



```
guest2@nguenziong:/tmp
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest@nguenziong tmp]$ echo "test" >> file01.txt
[guest@nguenziong tmp]$ chmod o+rx file01.txt
[guest@nguenziong tmp]$ ls -l file01.txt
-rw-rw-r-x. 1 guest guest 5 окт  6 11:37 file01.txt
[guest@nguenziong tmp]$ su guest2
Пароль:
[guest2@nguenziong tmp]$ cat file01.txt
test
[guest2@nguenziong tmp]$ echo "test" >> file01.txt
[guest2@nguenziong tmp]$ echo "test" > file01.txt
[guest2@nguenziong tmp]$ rm file01.txt
rm: невозможно удалить «file01.txt»: Операция не позволена
[guest2@nguenziong tmp]$ su
Пароль:
[root@nguenziong tmp]# chmod -t /tmp
[root@nguenziong tmp]# exit
exit
[guest2@nguenziong tmp]$ rm file01.txt
[guest2@nguenziong tmp]$ su
Пароль:
[root@nguenziong tmp]# chmod +t /tmp
[root@nguenziong tmp]# exit
exit
[guest2@nguenziong tmp]$
```

Figure 4: исследование Sticky-бита

Выводы

Изучили механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получили практические навыки работы в консоли с дополнительными атрибутами. Также мы рассмотрели работу механизма смены идентификатора процессов пользователей и влияние бита Sticky на запись и удаление файлов.