

Отчет по лабораторной работе №5

Жиронкин Павел владимирович НПИбд-01-18¹

Информационная Безопасность–2021, 10 ноября, 2021, Москва,
Россия

¹Российский Университет Дружбы Народов

Цели и задачи работы

Цель лабораторной работы

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

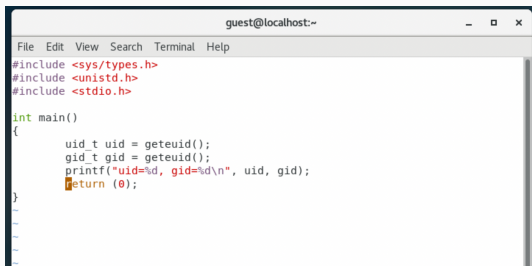
Задание к лабораторной работе

Лабораторная работа подразумевает выполнение последовательно необходимых действий, чтобы изучить механизмы изменения идентификаторов, применение SetUID- и Sticky-битов. Получить практические навыки работы в консоли с дополнительными атрибутами.

Процесс выполнения лабораторной работы

Процесс выполнения

1. Вошел в систему от имени пользователя guest, создал программу simpleid.c
2. Скомпилировал программу, выполнил ее. Выполнил системную программу id. И сравнил полученный результат с данными предыдущего пункта задания.
(рис. 1)



```
guest@localhost:~  
File Edit View Search Terminal Help  
#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. 1: Компиляция, выполнение программы

3. Усложнил программу, добавив вывод действительных идентификаторов.
4. Скомпилировал и запустил simpleid2.c. (рис. 2).

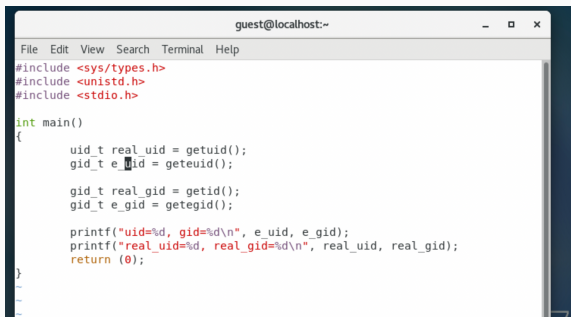
```
[guest@localhost ~]$ gcc simpleid.c -o simpleid
[guest@localhost ~]$ vim simpleid.c
[guest@localhost ~]$ ./simpleid
uid=1001, gid=1001
[guest@localhost ~]$ id
uid=1001(guest) gid=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@localhost ~]$
```

Рис. 2: Компиляция, выполнение программы

5. От имени суперпользователя выполнил команды:
`chown root:guest /home/guest/simpleid2; chmod u+s /home/guest/simpleid2.`
6. Выполнил проверку правильности установки новых атрибутов и смены владельца файла `simpleid2`: `ls -l simpleid2`. Запустила `simpleid2` и `id`.
7. Проделал тоже самое относительно SetGID-бита

Процесс выполнения

8. Создал программу readfile.c и откомпилировал ее.
9. Сменил владельца у файла readfile.c и изменил права так, чтобы только суперпользователь (root) мог прочитать его, а guest не мог. Проверил это. (рис. 3).



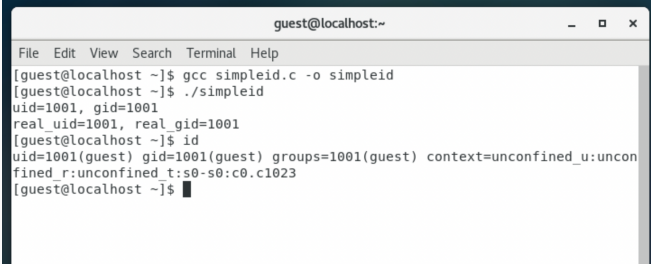
```
guest@localhost:~  
File Edit View Search Terminal Help  
#include <sys/types.h>  
#include <unistd.h>  
#include <stdio.h>  
  
int main()  
{  
    uid_t real_uid = getuid();  
    gid_t e_gid = geteuid();  
  
    gid_t real_gid = getid();  
    gid_t e_gid = getegid();  
  
    printf("uid=%d, gid=%d\n", e_uid, e_gid);  
    printf("real_uid=%d, real_gid=%d\n", real_uid, real_gid);  
    return (0);  
}
```

Рис. 3: Проверка

10. Сменил у программы readfile владельца и установил SetU'D-бит.
11. Проверил, может ли программа readfile прочитать файл readfile.c (может), проверил, может ли программа readfile прочитать файл /etc/shadow.

12. Выяснил, установлен ли атрибут Sticky на директории /tmp. От имени пользователя guest создал файл file01.txt в директории /tmp со словом test. Просмотрел атрибуты у только что созданного файла и разрешил чтение и запись для категории пользователей «все остальные».
13. От пользователя guest2 (не являющегося владельцем) попробовал прочитать файл /tmp/file01.txt, попробовал дозаписать в файл /tmp/file01.txt слово test2. Проверил содержимое файла. Также попробовал записать в файл /tmp/file01.txt слово test3, стеревав при этом всю имеющуюся в файле информацию. От пользователя guest2 попробовал удалить файл /tmp/file01.txt . (рис. 4).

Процесс выполнения

A terminal window titled 'guest@localhost:~' with standard window controls. The menu bar includes 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal shows the following commands and output:

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

Рис. 4: Выполнение и проверка от пользователя guest2

14. От суперпользователя выполнил команду, снимающую атрибут `t` (Sticky-бит) с директории `/tmp`.
15. От пользователя `guest2` проверил, что атрибута `t` у директории `/tmp` нет. Повторил предыдущие шаги. Нам удалось удалить файл от имени пользователя, не являющегося его владельцем, также получилось выполнить дозапись в файл и замену текста в файле.
16. От суперпользователя вернул атрибут `t` на директорию `/tmp`.

Выводы по проделанной работе

На основе проделанной работы изучил механизмы изменения идентификаторов, применение SetUID- и Sticky-битов. Получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работу механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.