Информационная безопасность лабораторная работа №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Ким Илья Владиславович НФИбд-01-21

Содержание

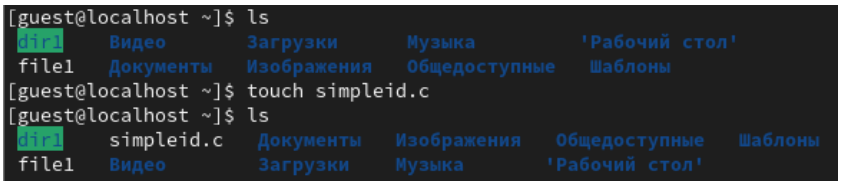
Цель работы

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

Создание программы

1. Создал программу simpleid.c (рис. [-@fig:001])

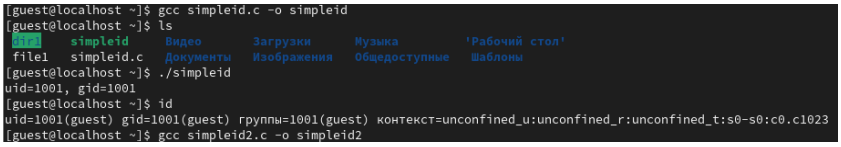


2. Записал код программы в simpleid.c (рис. [-@fig:002])

```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;
}
```

3. Скомпилировал и выполнил программу simpleid.c (рис. [-@fig:003])



4. Изменил код программы в simpleid.c (рис. [-@fig:004])

```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;
}
```

Получившуюся программу назовите simpleid2.c.

Скомпилируйте и запустите simpleid2.c:

```
gcc simpleid2.c -o simpleid2
./simpleid2
```

5. Скомпилировал и выполнил программу simpleid2.c (рис. [-@fig:005])

```
[guest@localhost ~]$ gcc simpleid.c -o simpleid2
[guest@localhost ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
```

6. От имени суперпользователя поменял права на файл simpleid2.c, поменял владельца файла и выполнил программу (рис. [-@fig:006])

```
[root@localhost guest]# chown root:guest /home/guest/simpleid2
[root@localhost guest]# chmod u+s /home/guest/simpleid2
[root@localhost guest]# ls -l simpleid2
-rwsr-xr-x. 1 root guest 17656 окт  5 18:46 simpleid2
[root@localhost guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@localhost guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@localhost guest]# touch readfile.c
[root@localhost guest]# ls
dir1         simpleid     Видео       Изображения    'Рабочий стол'
file1        simpleid2    Документы   Музыка          Шаблоны
readfile.c   simpleid.c   Загрузки    Общедоступные
```

- Результаты выполнения отличаются.

7. Создал файл readfile.c (рис. [-@fig:007])

```
[root@localhost guest]# touch readfile.c
[root@localhost guest]# ls
dir1         simpleid     Видео       Изображения    'Рабочий стол'
file1        simpleid2    Документы   Музыка          Шаблоны
readfile.c   simpleid.c   Загрузки    Общедоступные
```

8. Записал код программы в readfile.c (рис. [-@fig:008])

```
#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;
}
```

9. Скомпилировал readfile.c (рис. [-@fig:009])

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

10. Сменил права и владельца readfile.c, попрбовал от имени пользователя guest прочитать файл, получил отказ (рис. [-@fig:010])

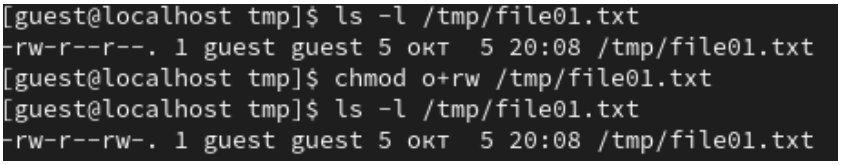
```
[root@localhost guest]# chown root:ilya readfile.c
[root@localhost guest]# ls -l readfile.c
-rw-r--r--. 1 root ilya 402 окт  5 19:28 readfile.c
[root@localhost guest]# chmod 700 readfile.c
[root@localhost guest]# ls -l readfile.c
-rwx------. 1 root ilya 402 окт  5 19:28 readfile.c
[root@localhost guest]# su guest
[guest@localhost ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
```

11. С помощью программы readfile прочитал файл /etc/shadow (рис. [-@fig:011])

Исследование Sticky-бита

1. Проверил установлен ли атрибут Sticky на директории /tmp , создал в ней файл file91.txt(рис. [-@fig:012])

2. Проверил атрибуты у файла file01.txt , поменял атрибуты на чтение и запись для остальных пользователей (рис. [-@fig:013])



3. От пользователя guest 2 попробовал прочитать, изменить и удалить file01.txt, получилось только прочитать (рис. [-@fig:014])

```
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ echo "test" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@localhost guest]$ rm /tmp/file01.txt
rm: удалить защищённый от записи обычный файл '/tmp/file01.txt'? y
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
```

4. От имени суперпользователя снял атрибут t у /tmp (рис. [-@fig:015])

```
[root@localhost guest]# chmod -t /tmp
```

5. От имени guest 2 проверил, что у директории /tmp нет атрибута t. Попробовал сделать команды еще раз, ничего не поменялось. (рис. [-@fig:016])

```
[guest2@localhost guest]$ ls -l / | grep tmp
drwxrwxrwx.  19 root root 4096 окт  5 20:12 tmp
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ echo "test" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
```

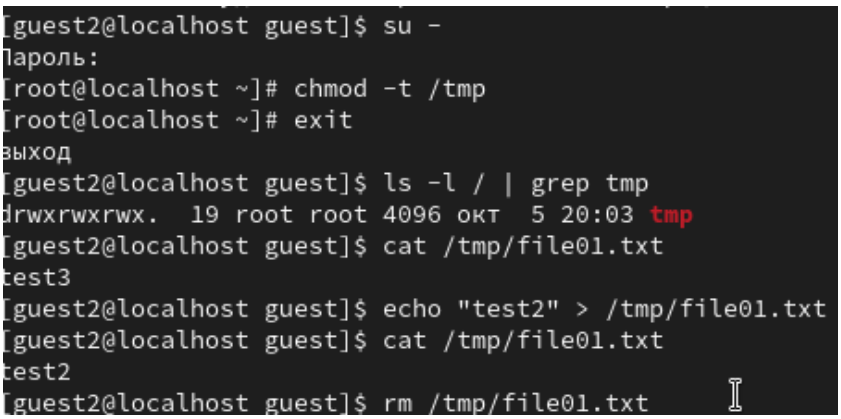
6. Поменял права файла file01.txt на 666 (рис. [-@fig:017])

```
[guest@localhost tmp]$ chmod 666 file01.txt
[guest@localhost tmp]$ ls -l file01.txt
```

7. Попробовал выполнить команды еще раз, получилось сделать все, кроме удаления (рис. [-@fig:018])

```
[guest2@localhost guest]$ echo "test2" > /tmp/file01.txt
[guest2@localhost guest]$ cat file01.txt
cat: file01.txt: Нет такого файла или каталога
[guest2@localhost guest]$ cat /tmp/file01.txt
test2
[guest2@localhost guest]$ echo "test3" > /tmp/file01.txt
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ rm /tmp/file01.txt
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
```

8. Удалил атрибут t и выполнил команды, получилось выполнить все, включая удаление (рис. [-@fig:019])


[guest2@localhost guest]$ su -
Пароль:
[root@localhost ~]# chmod -t /tmp
[root@localhost ~]# exit
выход
[guest2@localhost guest]$ ls -l / | grep tmp
drwxrwxrwx. 19 root root 4096 окт 5 20:03 tmp
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ echo "test2" > /tmp/file01.txt
[guest2@localhost guest]$ cat /tmp/file01.txt
test2
[guest2@localhost guest]$ rm /tmp/file01.txt

Выводы

Изучил механизм изменения идентификаторов, применяя SetUID и Sticky-битов. Получил практические навыки работы в консоли с дополнительными атрибутами. Рассмотрел работу механизма смены идентификатора процессов пользователей, а также влияния бить Sticky на запись и удаление файлов.