

Лабораторная работа №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

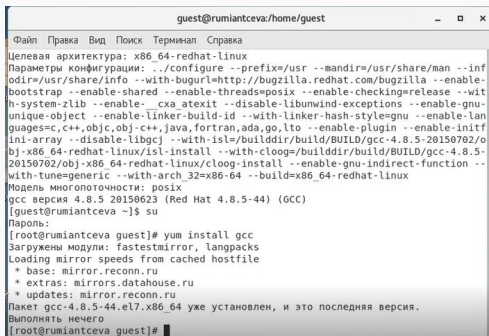
Румянцева Александра Сергеевна

12 ноября, 2021

Цель: 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.

Задание: Лабораторная работа подразумевает изучение влияния дополнительных атрибутов на файлы пользователя и изучение механизмов изменения идентификаторов.

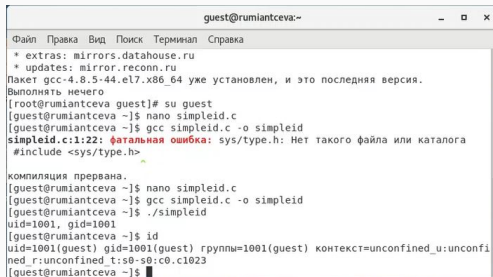
1. Проверила установлен ли компилятор gcc (рис. 1). В моём случае он уже установлен.



```
guest@rumiantceva:/home/guest
Файл Правка Вид Поиск Терминал Справка
Целевая архитектура: x86_64-redhat-linux
Параметры конфигурации: ./configure --prefix=/usr --mandir=/usr/share/man --inf
odir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-
bootstrap --enable-shared --enable-threads=posix --enable-checking=release --wit
h-system-zlib --enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-
unique-object --enable-linker-build-id --with-linker-hash-style=gnu --enable-lan
guages=c,c++,objc,obj-c++,java,fortran,ada,go,lto --enable-plugin --enable-initf
ini-array --disable-libgck --with-isl=/build/builddir/build/BUILD/gcc-4.8.5-20150702/o
bj-x86_64-redhat-linux/isl-install --with-cloog=/build/builddir/build/BUILD/gcc-4.8.5-
20150702/obj-x86_64-redhat-linux/cloog-install --enable-gnu-indirect-function --
with-tune=generic --with-arch_32=x86_64 --build=x86_64-redhat-linux
Модель многопоточности: posix
gcc версия 4.8.5 20150623 (Red Hat 4.8.5-44) (GCC)
[guest@rumiantceva ~]$ su
Пароль:
[root@rumiantceva guest]# yum install gcc
Загружены модули: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: mirror.reconn.ru
 * extras: mirrors.datahouse.ru
 * updates: mirror.reconn.ru
Пакет gcc-4.8.5-44.el7.x86_64 уже установлен, и это последняя версия.
Выполнять нечего
[root@rumiantceva guest]#
```

Figure 1: рис.1. Установка компилятора gcc.


2. Создала программу simpleid.c от имени пользователя guest, согласно работе.
- 3-5. Скомпилировала программу и убедилась, что файл программы создан командой `gcc simpleid.c -o simpleid`. Выполнила программу simpleid командой `./simpleid` и сравнила с выполнением команды `id`: пользователи и группы совпадают, при этом команда `id` вывела действительные идентификаторы, а программа вывела эффективные, но при этом они совпадают и выводят 1001, то есть пользователя guest (рис. 2).



```
guest@rumiantceva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
* extras: mirrors.datahouse.ru  
* updates: mirror.reconn.ru  
Пакет gcc-4.8.5-44.el7.x86_64 уже установлен, и это последняя версия.  
Выполнять нечего  
[root@rumiantceva guest]# su guest  
[guest@rumiantceva ~]$ nano simpleid.c  
[guest@rumiantceva ~]$ gcc simpleid.c -o simpleid  
simpleid.c:1:22: фатальная ошибка: sys/type.h: Нет такого файла или каталога  
#include <sys/type.h>  
компиляция прервана.  
[guest@rumiantceva ~]$ nano simpleid.c  
[guest@rumiantceva ~]$ gcc simpleid.c -o simpleid  
[guest@rumiantceva ~]$ ./simpleid  
uid=1001, gid=1001  
[guest@rumiantceva ~]$ id  
uid=1001(guest) gid=1001(guest) группы=1001(guest) контекст=unconfined_u:unconfi  
ned_r:unconfined_t:s0-s0:c0.c1023  
[guest@rumiantceva ~]$
```

Figure 2: рис.2. Компиляция программы simpleid, её выполнение и сравнение с командой `id`.

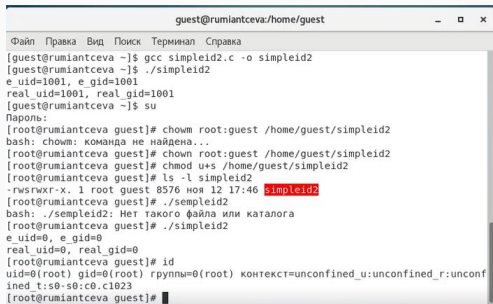
6. Усложнила программу, добавив вывод действительных идентификаторов, получившуюся программу назвала simpleid2.c.
7. Скомпилировала и запустила simpleid2.c командами `gcc simpleid2.c -o simpleid2` и `./simpleid2` (рис. 3). Видим, что программа выводит эффективные и действительные идентификаторы пользователя и группы для файла. Видим, что везде это 1001, то есть пользователь guest.



```
[guest@rumiantceva ~]$ nano simpleid.c
[guest@rumiantceva ~]$ gcc simpleid2.c -o simpleid2
[guest@rumiantceva ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real uid=1001, real gid=1001
[guest@rumiantceva ~]$
```

Figure 3: рис.3. Компиляция программы simpleid2, её выполнение.

8-11. От имени суперпользователя выполнила команды: `chown root:guest /home/guest/simpleid2` и `chmod u+s /home/guest/simpleid2`. Временно поменяв свои права с помощью `su` (рис. 4). С помощью этих команд файлу `simpleid2` изменила владельца и группу на `root` и `guest` соответственно (`chown`), а также установила на файл SetUID-бит (`chmod`). Выполнила проверку правильности установки новых атрибутов и смены владельца файла `simpleid2` командой `ls -l simpleid2`. Запустила `simpleid2` и `id` командами `./simpleid2` и `id`. Сравнила результаты: действительные идентификаторы совпадают с выводом команды `id` - везде 0, то есть рут-пользователь. Так же важно заметить, что эффективные идентификаторы совпадают с действительными.



```
guest@rumiantceva:/home/guest
Файл Правка Вид Поиск Терминал Справка
[guest@rumiantceva ~]$ gcc simpleid2.c -o simpleid2
[guest@rumiantceva ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@rumiantceva ~]$ su
Пароль:
[root@rumiantceva guest]# chown root:guest /home/guest/simpleid2
bash: chown: команда не найдена...
[root@rumiantceva guest]# chown root:guest /home/guest/simpleid2
[root@rumiantceva guest]# chmod u+s /home/guest/simpleid2
[root@rumiantceva guest]# ls -l simpleid2
-rwsrwxr-x. 1 root guest 8576 ноя 12 17:46 simpleid2
[root@rumiantceva guest]# ./simpleid2
bash: ./simpleid2: Нет такого файла или каталога
[root@rumiantceva guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@rumiantceva guest]# id
uid=0(root) gid=0(root) rpyunny=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@rumiantceva guest]#
```

Figure 4: рис.4. Изменение владельца программы и установка SetUID-бита, проверка установки и изменения, запуск программы и команды `id`.

12. Прodelала тоже самое относительно SetGID-бита (рис. 5)

Установка SetGID-бита отражается к команде `ls`, а сравнение выполнения программы и команды `id` дало следующие результаты: действительные идентификаторы совпадают с выводом команды `id` - везде 0, то есть рут-пользователь. Но так же важно заметить, что эффективные идентификаторы отличны от действительных: пользователь - 0, группа - 1001.

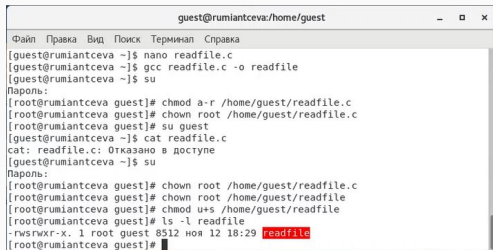
A terminal window showing the installation and verification of the SetGID bit. The user is in a directory /home/guest/simpleid2. They run 'chmod g+s /home/guest/simpleid2'. Then they run 'ls -l simpleid2', which shows a file with permissions '-rwsrwsr-x' and the SetGID bit set (indicated by a red 's' in the permissions). They then run './simpleid2', which outputs 'e_uid=0, e_gid=1001'. Finally, they run 'id', which shows they are root (uid=0, gid=0) in an unconfined context.

```
[root@rumiantceva guest]# chmod g+s /home/guest/simpleid2
[root@rumiantceva guest]# ls -l simpleid2
-rwsrwsr-x. 1 root guest 8576 ноя 12 17:46 simpleid2
[root@rumiantceva guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
[root@rumiantceva guest]# id
uid=0(root) gid=0(root) rpyyny=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@rumiantceva guest]#
```

Figure 5: рис.5. Установка SetGID-бита, проверка установки, запуск программы и команды `id`.

13-14 Создала программу readfile.c в соответствии с заданием. Откомпилировала её командой gcc readfile.c -o readfile.

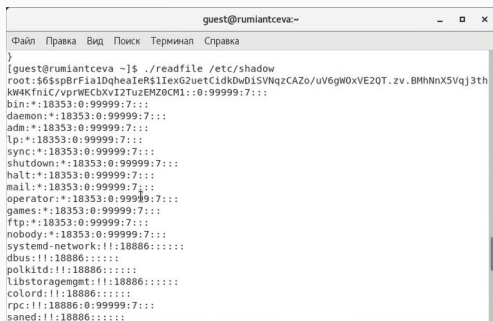
15-17. Сменила владельца у файла readfile.c (chown) и изменила права так, чтобы только суперпользователь (root) мог прочитать его, а guest не мог. Использовала chmod a-r. Проверила, что пользователь guest не может прочитать файл readfile.c командой cat. Сменила у программы readfile владельца и установила SetUID-бит (рис. 6).



```
guest@rumiantceva:/home/guest
Файл Правка Вид Поиск Терминал Справка
[guest@rumiantceva ~]$ nano readfile.c
[guest@rumiantceva ~]$ gcc readfile.c -o readfile
[guest@rumiantceva ~]$ su
Пароль:
[root@rumiantceva guest]# chmod a-r /home/guest/readfile.c
[root@rumiantceva guest]# chown root /home/guest/readfile.c
[root@rumiantceva guest]# su guest
[guest@rumiantceva ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
[guest@rumiantceva ~]$ su
Пароль:
[root@rumiantceva guest]# chown root /home/guest/readfile.c
[root@rumiantceva guest]# chown root /home/guest/readfile
[root@rumiantceva guest]# chmod u+s /home/guest/readfile
[root@rumiantceva guest]# ls -l readfile
-rwsrwxr-x. 1 root guest 8512 ноя 12 18:29 readfile
[root@rumiantceva guest]#
```

Figure 6: рис.6. Компиляция readfile и другие действия в соответствии с 14-17 пунктами.

18- 19. Проверила, может ли программа readfile прочитать файл readfile.c.
Проверила, может ли программа readfile прочитать файл /etc/shadow (рис. 7).



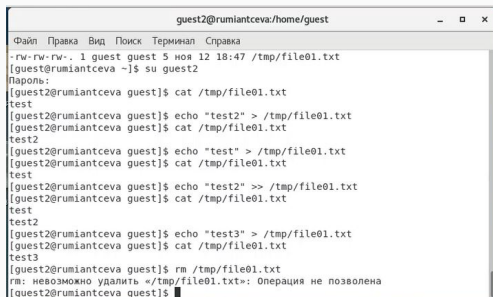
```
guest@rumiantceva:~  
Файл  Правка  Вид  Поиск  Терминал  Справка  
}  
[guest@rumiantceva ~]$ ./readfile /etc/shadow  
root:$6$spBrFia1DqheaIeR$1IexG2uetC1dkDwD1SVNqzCAZo/uV6gW0xVE2QT.zv.BMhNnx5Vqj3th  
kW4KfniC/vprWEcbXvI2TuzEMZ0CM1::0:99999:7:::  
bin:!:18353:0:99999:7:::  
daemon:!:18353:0:99999:7:::  
adm:!:18353:0:99999:7:::  
lp:!:18353:0:99999:7:::  
sync:!:18353:0:99999:7:::  
shutdown:!:18353:0:99999:7:::  
halt:!:18353:0:99999:7:::  
mail:!:18353:0:99999:7:::  
operator:!:18353:0:99999:7:::  
games:!:18353:0:99999:7:::  
ftp:!:18353:0:99999:7:::  
nobody:!:18353:0:99999:7:::  
systemd-network:!!:18886:::  
dbus:!!:18886:::  
polkitd:!!:18886:::  
libstoragemgmt:!!:18886:::  
colord:!!:18886:::  
rpc:!!:18886:0:99999:7:::  
saned:!!:18886:::
```

Figure 7: рис.7. Выполнение программы readfile с файлом /etc/shadow.

1-9. От имени пользователя guest создала файл file01.txt в директории /tmp со словом test. Просмотрела атрибуты у только что созданного файла и разрешила чтение и запись для категории пользователей «все остальные»

От пользователя guest2 (не являющегося владельцем) попробовала прочитать файл /tmp/file01.txt командой cat /tmp/file01.txt (рис. 12). От пользователя guest2 попробовала дозаписать в файл слово test2 и записать в файл слово test3 (рис. 8). Действия удалось выполнить.

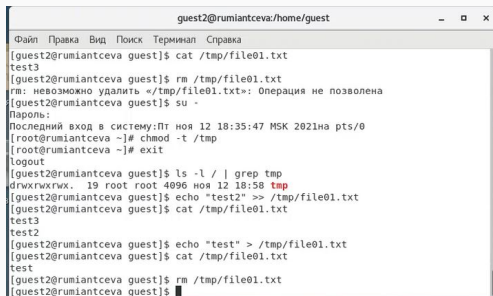
От пользователя guest2 попробовала удалить файл /tmp/file01.txt. Мне не удалось удалить файл (рис. 8).



```
guest2@rumiantceva:/home/guest
Файл  Правка  Вид  Поиск  Терминал  Справка
-rw-rw-rw-. 1 guest guest 5 ноя 12 18:47 /tmp/file01.txt
[guest@rumiantceva ~]$ su guest2
Пароль:
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test
[guest2@rumiantceva guest]$ echo "test2" > /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test2
[guest2@rumiantceva guest]$ echo "test" > /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test
[guest2@rumiantceva guest]$ echo "test2" >> /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test
test2
[guest2@rumiantceva guest]$ echo "test3" > /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test3
[guest2@rumiantceva guest]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: Операция не позволена
[guest2@rumiantceva guest]$
```

Figure 8: рис.8. Выполнение пунктов 5-9 исследования Sticky-бита .

10-14. Сняла атрибут t (Sticky-бит) с директории /tmp с помощью root-пользователя. Повторила предыдущие шаги. Видим, что дозапись и запись так же разрешены, но при этом удалось и удалить файл.



```
guest2@rumiantceva:/home/guest
Файл  Правка  Вид  Поиск  Терминал  Справка
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test3
[guest2@rumiantceva guest]$ rm /tmp/file01.txt
rm: невозможно удалить «/tmp/file01.txt»: операция не позволена
[guest2@rumiantceva guest]$ su -
Пароль:
Последний вход в систему: Пт ноя 12 18:35:47 MSK 2021 на pts/0
[root@rumiantceva ~]# chmod -t /tmp
[root@rumiantceva ~]# exit
logout
[guest2@rumiantceva guest]$ ls -l / | grep tmp
drwxrwxrwx. 19 root root 4096 ноя 12 18:58 tmp
[guest2@rumiantceva guest]$ echo "test2" >> /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test3
test2
[guest2@rumiantceva guest]$ echo "test" > /tmp/file01.txt
[guest2@rumiantceva guest]$ cat /tmp/file01.txt
test
[guest2@rumiantceva guest]$ rm /tmp/file01.txt
[guest2@rumiantceva guest]$
```

Figure 9: рис.9. Выполнение пунктов 10-13 исследования Sticky-бита .

15. Повысила свои права до суперпользователя и вернула атрибут t на директорию /tmp: su -, chmod +t /tmp, exit.a.

Я изучила механизмы изменения идентификаторов, применения SetUID- и Sticky-битов. Получила практические навыки работы в консоли с дополнительными атрибутами. Рассмотрела работу механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.