

Отчёт по лабораторной работе №5, Информационная безопасность

Дискреционное разграничение прав в Linux. Исследование влияния
дополнительных атрибутов

Выполнила: Ганина Таисия Сергеевна, НКАбд-01-22

Содержание

1	Цель работы	5
2	Теоретическое введение	6
3	Выполнение лабораторной работы	8
4	Вывод	14
5	Список литературы. Библиография	15

Список иллюстраций

3.1	8
3.2	8
3.3	9
3.4	9
3.5	9
3.6	10
3.7	10
3.8	11
3.9	11
3.10	11
3.11	12
3.12	12
3.13	12
3.14	12
3.15	12
3.16	13
3.17	13
3.18	13

Список таблиц

1 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

2 Теоретическое введение

1. Дополнительные атрибуты файлов Linux

В Linux существует три основных вида прав — право на чтение (read), запись (write) и выполнение (execute), а также три категории пользователей, к которым они могут применяться — владелец файла (user), группа владельца (group) и все остальные (others). Но, кроме прав чтения, выполнения и записи, есть еще три дополнительных атрибута. [1]

- **Sticky bit**

Используется в основном для каталогов, чтобы защитить в них файлы. В такой каталог может писать любой пользователь. Но, из такой директории пользователь может удалить только те файлы, владельцем которых он является. Примером может служить директория /tmp, в которой запись открыта для всех пользователей, но нежелательно удаление чужих файлов.

- **SUID (Set User ID)**

Атрибут исполняемого файла, позволяющий запустить его с правами владельца. В Linux приложение запускается с правами пользователя, запустившего указанное приложение. Это обеспечивает дополнительную безопасность т.к. процесс с правами пользователя не сможет получить доступ к важным системным файлам, которые принадлежат пользователю root.

- **SGID (Set Group ID)**

Аналогичен `suid`, но относиться к группе. Если установить `sgid` для каталога, то все файлы созданные в нем, при запуске будут принимать идентификатор группы каталога, а не группы владельца, который создал файл в этом каталоге.

- **Обозначение атрибутов `sticky`, `suid`, `sgid`**

Специальные права используются довольно редко, поэтому при выводе программы `ls -l` символ, обозначающий указанные атрибуты, закрывает символ стандартных прав доступа.

Пример:

```
rwsrwsrwt
```

где первая s — это `suid`, вторая s — это `sgid`, а последняя t — это `sticky bit`

В приведенном примере не понятно, `rwt` — это `rw`- или `rwX`? Определить это просто. Если `t` маленькое, значит `x` установлен. Если `T` большое, значит `x` не установлен. То же самое правило распространяется и на `s`.

В числовом эквиваленте данные атрибуты определяются первым символом при четырехзначном обозначении (который часто опускается при назначении прав), например в правах `1777` — символ `1` обозначает `sticky bit`. Остальные атрибуты имеют следующие числовое соответствие:

1 — установлен `sticky bit`

2 — установлен `sgid`

4 — установлен `suid`

2. Компилятор GCC

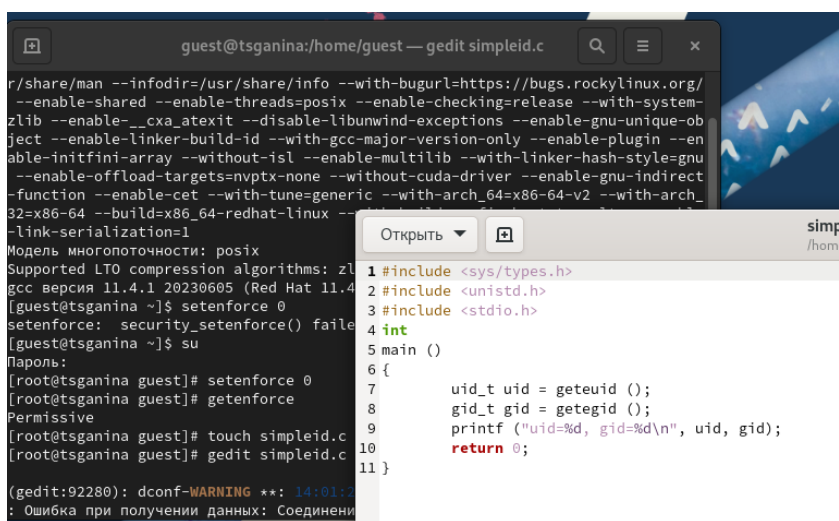
`GCC` - это свободно доступный оптимизирующий компилятор для языков `C`, `C++`. Собственно программа `gcc` это некоторая надстройка над группой компиляторов, которая способна анализировать имена файлов, передаваемые ей в качестве аргументов, и определять, какие действия необходимо выполнить. Файлы с расширением `.cc` или `.C` рассматриваются, как файлы на языке `C++`, файлы с расширением `.c` как программы на языке `C`, а файлы с расширением `.o` считаются объектными. [2]

3 Выполнение лабораторной работы

Выполнение заданий по лабораторной работе (рис. 3.1, 3.2, 3.3, 3.4, 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18).

```
[guest@tsganina ~]$ su
Пароль:
[root@tsganina guest]# setenforce 0
[root@tsganina guest]# getenforce
Permissive
[root@tsganina guest]#
```

Рис. 3.1:



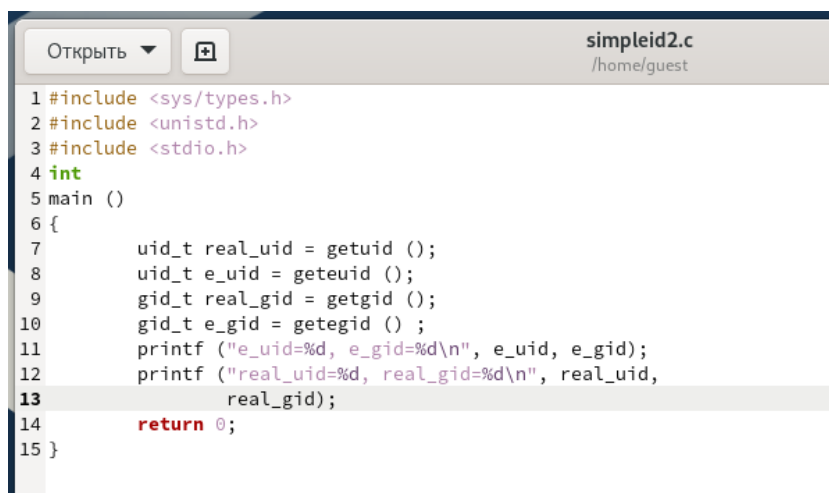
```
guest@tsganina:/home/guest — gedit simpleid.c
r/share/man --infodir=/usr/share/info --with-bugurl=https://bugs.rockylinux.org/
--enable-shared --enable-threads=posix --enable-checking=release --with-system-
zlib --enable-__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-ob
ject --enable-linker-build-id --with-gcc-major-version-only --enable-plugin --en
able-initfini-array --without-isl --enable-multilib --with-linker-hash-style=gnu
--enable-offload-targets=nvptx-none --without-cuda-driver --enable-gnu-indirect
-function --enable-cet --with-tune=generic --with-arch_64=x86-64-v2 --with-arch_
32=x86-64 --build=x86_64-redhat-linux --with-arch_32=x86_64-v1 --with-arch_64=
--link-serialization=1
Модель многопоточности: posix
Supported LTO compression algorithms: zli
gcc версия 11.4.1 20230605 (Red Hat 11.4
[guest@tsganina ~]$ setenforce 0
setenforce: security_setenforce() failed
[guest@tsganina ~]$ su
Пароль:
[root@tsganina guest]# setenforce 0
[root@tsganina guest]# getenforce
Permissive
[root@tsganina guest]# touch simpleid.c
[root@tsganina guest]# gedit simpleid.c
(gedit:92280): dconf-WARNING **: 14:01:1
: Ошибка при получении данных: Соединени
```

```
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t uid = geteuid ();
8     gid_t gid = getegid ();
9     printf ("uid=%d, gid=%d\n", uid, gid);
10    return 0;
11 }
```

Рис. 3.2:

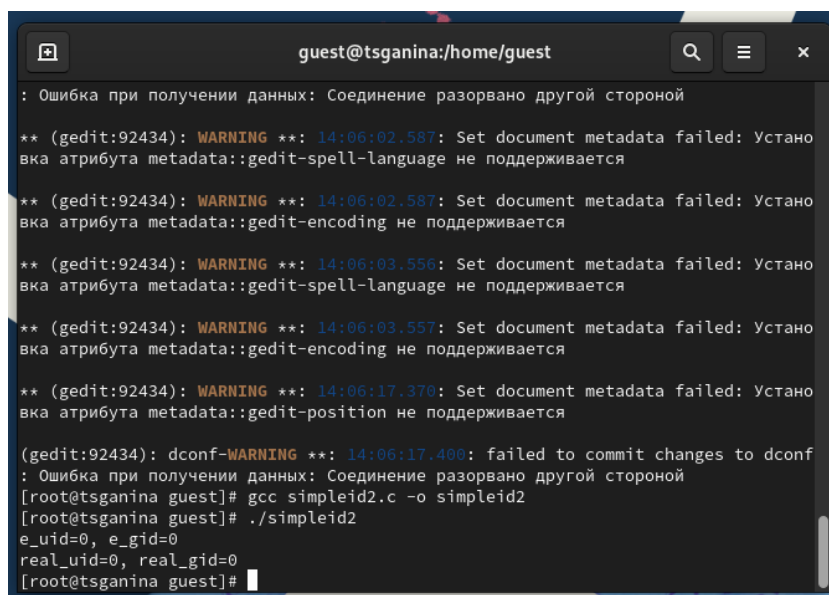

```
[root@tsganina guest]# gcc simpleid.c -o simpleid
[root@tsganina guest]# ./simpleid
uid=0, gid=0
[root@tsganina guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@tsganina guest]#
```

Рис. 3.3:



```
Открыть simpleid2.c /home/guest
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int
5 main ()
6 {
7     uid_t real_uid = getuid ();
8     uid_t e_uid = geteuid ();
9     gid_t real_gid = getgid ();
10    gid_t e_gid = getegid ();
11    printf ("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
12    printf ("real_uid=%d, real_gid=%d\n", real_uid,
13           real_gid);
14    return 0;
15 }
```

Рис. 3.4:



```
guest@tsganina:/home/guest
: Ошибка при получении данных: Соединение разорвано другой стороной
** (gedit:92434): WARNING **: 14:06:02.587: Set document metadata failed: Установка атрибута metadata::gedit-spell-language не поддерживается
** (gedit:92434): WARNING **: 14:06:02.587: Set document metadata failed: Установка атрибута metadata::gedit-encoding не поддерживается
** (gedit:92434): WARNING **: 14:06:03.556: Set document metadata failed: Установка атрибута metadata::gedit-spell-language не поддерживается
** (gedit:92434): WARNING **: 14:06:03.557: Set document metadata failed: Установка атрибута metadata::gedit-encoding не поддерживается
** (gedit:92434): WARNING **: 14:06:17.370: Set document metadata failed: Установка атрибута metadata::gedit-position не поддерживается
(gedit:92434): dconf-WARNING **: 14:06:17.400: failed to commit changes to dconf
: Ошибка при получении данных: Соединение разорвано другой стороной
[root@tsganina guest]# gcc simpleid2.c -o simpleid2
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@tsganina guest]#
```

Рис. 3.5:

```
guest@tsganina:/home/guest
** (gedit:92434): WARNING **: 14:06:17.370: Set document metadata failed: Установка атрибута metadata::gedit-position не поддерживается

(gedit:92434): dconf-WARNING **: 14:06:17.400: failed to commit changes to dconf
: Ошибка при получении данных: Соединение разорвано другой стороной
[root@tsganina guest]# gcc simpleid2.c -o simpleid2
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@tsganina guest]# chown root:guest /home/guest/simpleid2
[root@tsganina guest]# chmod u+s /home/guest/simpleid2
[root@tsganina guest]# ls -l simpleid2
-rwsr-xr-x. 1 root guest 26064 anp 12 14:06 simpleid2
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@tsganina guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@tsganina guest]# chown root:guest simpleid2
[root@tsganina guest]# chmod g+s simpleid2
[root@tsganina guest]# ls -l simpleid2
-rwxr-sr-x. 1 root guest 26064 anp 12 14:06 simpleid2
[root@tsganina guest]#
```

Рис. 3.6:

```
guest@tsganina:/home/guest
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@tsganina guest]# chown root:guest /home/guest/simpleid2
[root@tsganina guest]# chmod u+s /home/guest/simpleid2
[root@tsganina guest]# ls -l simpleid2
-rwsr-xr-x. 1 root guest 26064 anp 12 14:06 simpleid2
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@tsganina guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@tsganina guest]# chown root:guest simpleid2
[root@tsganina guest]# chmod g+s simpleid2
[root@tsganina guest]# ls -l simpleid2
-rwxr-sr-x. 1 root guest 26064 anp 12 14:06 simpleid2
[root@tsganina guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
[root@tsganina guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[root@tsganina guest]#
```

Рис. 3.7:

```
[root@tsganina guest]# gcc readfile.c -o readfile
[root@tsganina guest]# chown root:guest readfile
[root@tsganina guest]# chmod 700 readfile
[root@tsganina guest]# chown root:guest readfile
[root@tsganina guest]# chmod -r readfile.c
[root@tsganina guest]# chmod u+s readfile
[root@tsganina guest]# exit
exit
[guest@tsganina ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
[guest@tsganina ~]$
```

Рис. 3.8:

```
[root@tsganina guest]# exit
exit
[guest@tsganina ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
[guest@tsganina ~]$ ./readfile readfile.c
bash: ./readfile: Отказано в доступе
[guest@tsganina ~]$ ./readfile /etc/shadow
bash: ./readfile: Отказано в доступе
[guest@tsganina ~]$
```

Рис. 3.9:

```
guest@tsganina:/home/guest
}
[root@tsganina guest]# ./readfile readfile.c
#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;
}
[root@tsganina guest]#
```

Рис. 3.10:

```
[guest@tsganina ~]$ ls -l / | grep tmp
drwxrwxrwt. 18 root root 4096 anp 12 14:33 tmp
[guest@tsganina ~]$ echo "test" > /tmp/file01.txt
[guest@tsganina ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 anp 12 14:35 /tmp/file01.txt
[guest@tsganina ~]$ chmod o+rw /tmp/file01.txt
[guest@tsganina ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 anp 12 14:35 /tmp/file01.txt
[guest@tsganina ~]$
```

Рис. 3.11:

```
[guest@tsganina ~]$ su guest2
Пароль:
[guest2@tsganina guest]$ cat /tmp/file01.txt
test
[guest2@tsganina guest]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Нет такого файла или каталога
[guest2@tsganina guest]$ echo "test2" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$
```

Рис. 3.12:

```
[guest2@tsganina guest]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$
```

Рис. 3.13:

```
[guest2@tsganina guest]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$ cat /tmp/file01.txt
test
[guest2@tsganina guest]$ rm /tmp/file01.txt
rm: удалить защищенный от записи обычный файл '/tmp/file01.txt'? y
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
[guest2@tsganina guest]$
```

Рис. 3.14:

```
[guest2@tsganina guest]$ su -
Пароль:
[root@tsganina ~]# chmod -t /tmp
[root@tsganina ~]# exit
выход
[guest2@tsganina guest]$ ls -l / | grep tmp
drwxrwxrwx. 18 root root 4096 anp 12 14:42 tmp
[guest2@tsganina guest]$
```

Рис. 3.15:

```
[guest2@tsganina guest]$ cat /tmp/file01.txt
test
[guest2@tsganina guest]$ echo "test2" >> /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$ echo "test3" > /tmp/file01.txt
bash: /tmp/file01.txt: Отказано в доступе
[guest2@tsganina guest]$ rm /tmp/file01.txt
rm: удалить защищенный от записи обычный файл '/tmp/file01.txt'? y
```

Рис. 3.16:

```
[guest2@tsganina guest]$ ls /tmp
HcRFYpzV4M
systemd-private-1d31b221501f43f98e958d52ecc570d0-chronyd.service-BHxbxq
systemd-private-1d31b221501f43f98e958d52ecc570d0-colord.service-xPKNig
systemd-private-1d31b221501f43f98e958d52ecc570d0-dbus-broker.service-LL4Ry4
systemd-private-1d31b221501f43f98e958d52ecc570d0-fwupd.service-01TZE5
systemd-private-1d31b221501f43f98e958d52ecc570d0-ModemManager.service-vkWILX
systemd-private-1d31b221501f43f98e958d52ecc570d0-power-profiles-daemon.service-B
3mCws
systemd-private-1d31b221501f43f98e958d52ecc570d0-rtkit-daemon.service-w12cBn
systemd-private-1d31b221501f43f98e958d52ecc570d0-switcheroo-control.service-6opq
QN
systemd-private-1d31b221501f43f98e958d52ecc570d0-systemd-logind.service-oT0cqP
systemd-private-1d31b221501f43f98e958d52ecc570d0-upower.service-wqabHK
Temp-432273a9-39df-45ee-aafa-7401bee62e20
tmpaddon
tmpaddon-1
tmpaddon-2
vboxguest-Module.symvers
[guest2@tsganina guest]$
```

Рис. 3.17:

```
vboxguest-Module.symvers
[guest2@tsganina guest]$ su -
Пароль:
[root@tsganina ~]# chmod +t /tmp
[root@tsganina ~]# exit
выход
[guest2@tsganina guest]$
```

Рис. 3.18:

4 Вывод

Были изучены механизмы изменения идентификаторов и применения SetUID- и Sticky-битов. Получены практические навыки работы в консоли с дополнительными атрибутами. Были рассмотрены работа механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов

5 Список литературы. Библиография

[0] Методические материалы курса

[1] Дополнительные атрибуты: <https://tokmakov.msk.ru/blog/item/141>

[2] Компилятор GSS: <http://parallel.imm.uran.ru/freesoft/make/instrum.html>