Защита лабораторной работы №5. Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

Смородова Дарья Владимировна

2022 Oct 8th

RUDN University, Moscow, Russian Federation

Цель выполнения лабораторной работы

Цель выполнения 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.

Результаты выполнения лабораторной работы

Проверка установки компилятора gcc

```
[smorodovadv@smorodovadv ~]$ gcc -v
Используются внутренние спецификации.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/11/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Целевая архитектура: x86_64-redhat-linux
Параметры конфигурации: ../configure --enable-bootstrap --enable-host-pie --enab
le-host-bind-now --enable-languages=c,c++,fortran,lto --prefix=/usr --mandir=/us
r/share/man --infodir=/usr/share/info --with-bugurl=https://bugs.rockylinux.org/
 --enable-shared --enable-threads=posix --enable-checking=release --enable-multi
lib --with-system-zlib --enable-__cxa_atexit --disable-libunwind-exceptions --en
able-gnu-unique-object --enable-linker-build-id --with-gcc-major-version-only --
with-linker-hash-style=gnu --enable-plugin --enable-initfini-array --without-isl
 --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-build-config=bootstrap-lto --enable
-link-serialization=1
Модель многопоточности: posix
Supported LTO compression algorithms: zlib zstd
gcc версия 11.2.1 20220127 (Red Hat 11.2.1-9) (GCC)
[smorodovadv@smorodovadv ~]$ yum install gcc
Ошибка: Эту команду нужно запускать с привилегиями суперпользователя (на большин
стве систем - под именем пользователя root).
[smorodovadv@smorodovadv ~]$ su
Пароль:
[root@smorodovadv smorodovadv]# yum install gcc
Последняя проверка окончания срока действия метаданных: 0:32:53 назад, Пт 30 сен
 2022 17:20:18.
Пакет gcc-11.2.1-9.4.el9.x86_64 уже установлен.
Зависимости разрешены.
Отсутствуют действия для выполнения.
Выполнено!
```

Figure 1: Проверка установки компилятора gcc

Отключение системы запретов

```
[root@smorodovadv smorodovadv]# setenforce 0
[root@smorodovadv smorodovadv]# getenforce
Permissive
```

Figure 2: Отключение системы запретов

Проверка компиляторов

```
[root@smorodovadv smorodovadv]# whereis gcc
gcc: /usr/bin/gcc /usr/lib/gcc /usr/libexec/gcc /usr/share/man/man1/gcc.1.gz /us
r/share/info/gcc.info.gz
[root@smorodovadv smorodovadv]# whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
[root@smorodovadv smorodovadv]#
```

Figure 3: Проверка компиляторов

Содержимое файла simpleid.c

```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;
}
```

Figure 4: Содержимое файла simpleid.c

Компиляция и запуск файла simpleid.c

```
[guest@smorodovadv ~]$ gcc simpleid.c -o simpleid
[guest@smorodovadv ~]$ ./simpleid
uid=1001, gid=1001
```

Figure 5: Компиляция и запуск файла simpleid.c

Команда id

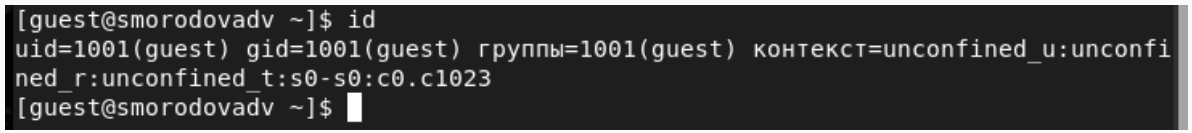

Figure 6: Команда id

Содержимое файла simpleid2.c

```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;
}
```

Figure 7: Содержимое файла simpleid2.c

Компиляция и запуск файла simpleid2.c

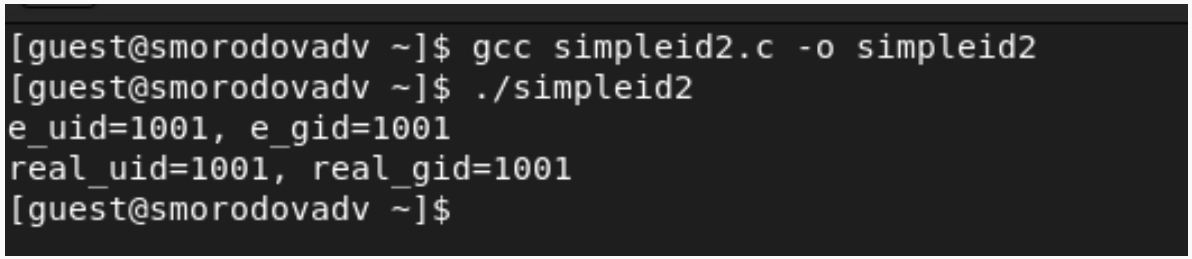

Figure 8: Компиляция и запуск файла simpleid2.c

Команды chown и chmod

```
[guest@smorodovadv ~]$ su
Пароль:
[root@smorodovadv guest]# chown root:guest /home/guest/simpleid2
[root@smorodovadv guest]# chmod u+s /home/guest/simpleid2
[root@smorodovadv guest]#
```

Figure 9: Команды chown и chmod

Проверка правильности установки новых атрибутов

```
[root@smorodovadv guest]# ls -l simpleid2
-rwsrwxr-x. 1 root guest 26008 сен 30 18:05 simpleid2
[root@smorodovadv guest]#
```

Figure 10: Проверка правильности установки новых атрибутов

Запуск simpleid2 и id

```
[root@smorodovadv guest]# ./simpleid2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[root@smorodovadv guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconf
ined_t:s0-s0:c0.c1023
[root@smorodovadv guest]#
```

Figure 11: Запуск simpleid2 и id

Сравнение SetGID-бита

```
[root@smorodovadv guest]# chmod g+s /home/guest/simpleid2
[root@smorodovadv guest]# ls -l simpleid2
-rwsrwsr-x. 1 root guest 26008 сен 30 18:05 simpleid2
[root@smorodovadv guest]# ./simpleid2
e_uid=0, e_gid=1001
real_uid=0, real_gid=0
[root@smorodovadv guest]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconf
ined_t:s0-s0:c0.c1023
[root@smorodovadv guest]#
```

Figure 12: Сравнение SetGID-бита

Содержание файла readfile.c

```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;
}
```

Figure 13: Содержание файла readfile.c

Компиляция файла readfile.c

```
[root@smorodovadv guest]# su guest
[guest@smorodovadv ~]$ touch readfile.c
[guest@smorodovadv ~]$ gcc readfile.c -o readfile
[guest@smorodovadv ~]$
```

Figure 14: Компиляция файла readfile.c

Смена владельца и прав у файла readfile.c

```
[guest@smorodovadv ~]$ su
Пароль:
[root@smorodovadv guest]# chown root:guest readfile.c
[root@smorodovadv guest]# chmod 700 readfile.c
```

Figure 15: Смена владельца и прав у файла readfile.c

Проверка возможности прочитать файл readfile.c

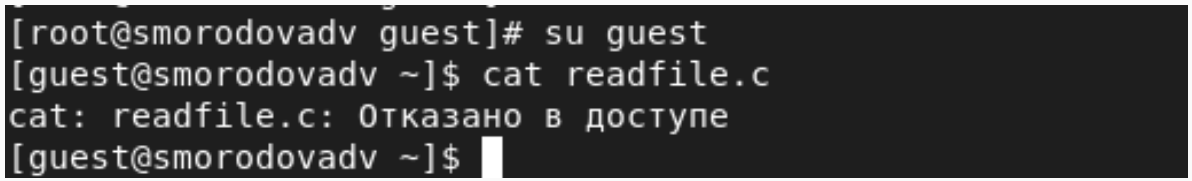

Figure 16: Проверка возможности прочитать файл readfile.c

Смена у программы readfile владельца и установка SetU’D-бит

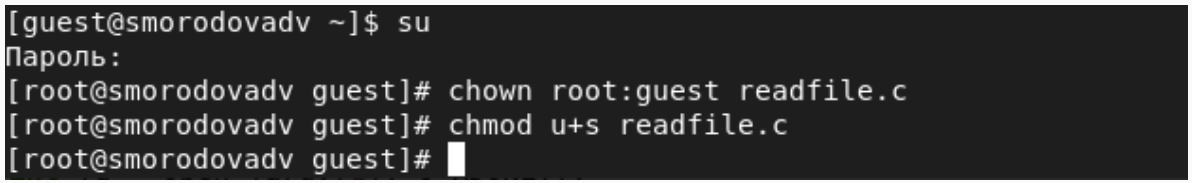

Figure 17: Смена у программы readfile владельца и установка SetU’D-бит

Проверка возможности прочитать файл readfile.c

```
[root@smorodovadv 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@smorodovadv guest]#
```

Figure 18: Проверка возможности прочитать файл readfile.c

Проверка возможности прочитать файл /etc/shadow

Figure 19: Проверка возможности прочитать файл /etc/shadow

Проверка установки атрибута Sticky на директории /tmp

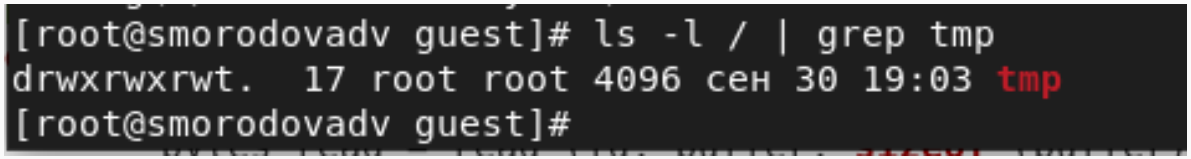

Figure 20: Проверка установки атрибута Sticky на директории /tmp

Создание файла file01.txt в директории /tmp со словом test

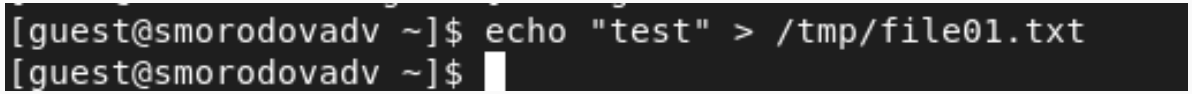

Figure 21: Создание файла file01.txt в директории /tmp со словом test

Просмотр атрибутов и разрешение чтения и записи для категории пользователей «все остальные»

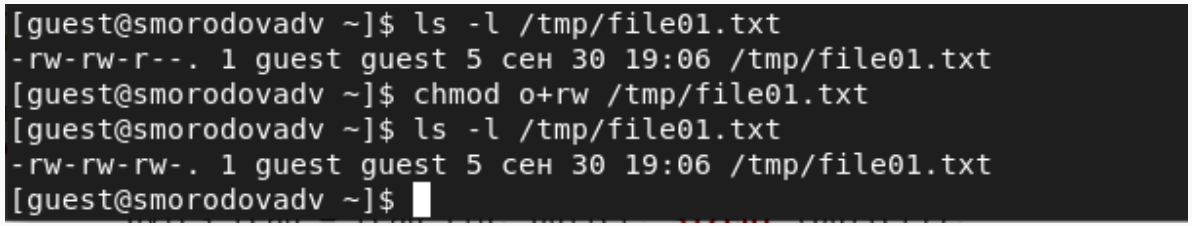

Figure 22: Просмотр атрибутов и разрешение чтения и записи для категории пользователей «все остальные»

Попытка прочитать файл /tmp/file01.txt от пользователя guest2

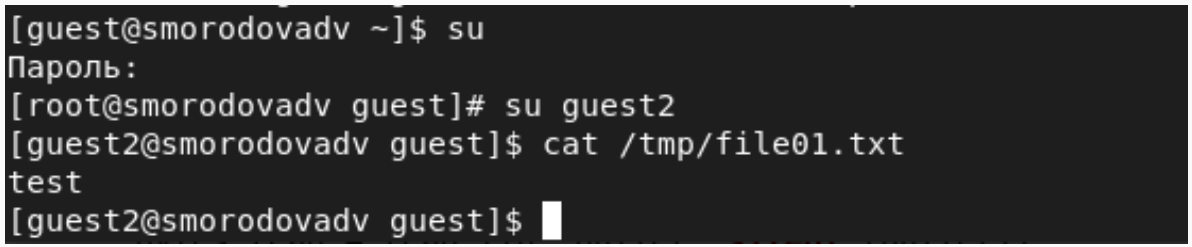

Figure 23: Попытка прочитать файл /tmp/file01.txt от пользователя guest2

Попытка дозаписать в файл /tmp/file01.txt слово test2 от пользователя guest2

Figure 24: Попытка дозаписать в файл /tmp/file01.txt слово test2 от пользователя guest2

Попытка перезаписать файл /tmp/file01.txt словом test3 от пользователя guest2

```
[guest2@smorodovadv guest]$ echo "test3"> /tmp/file01.txt
[guest2@smorodovadv guest]$ cat /tmp/file01.txt
test3
[guest2@smorodovadv guest]$
```

Figure 25: Попытка перезаписать файл /tmp/file01.txt словом test3 от пользователя guest2

Попытка удалить файл /tmp/file01.txt от пользователя guest2

Figure 26: Попытка удалить файл /tmp/file01.txt от пользователя guest2

Снятие атрибута t с директории /tmp

```
[guest2@smorodovadv guest]$ su -
Пароль:
[root@smorodovadv ~]# chmod -t /tmp
[root@smorodovadv ~]# exit
выход
[guest2@smorodovadv guest]$
```

Figure 27: Снятие атрибута t с директории /tmp

Проверка снятия атрибута t с директории /tmp

```
[guest2@smorodovadv guest]$ ls -l / | grep tmp
drwxrwxrwx.  17 root root 4096 сен 30 19:14 tmp
[guest2@smorodovadv guest]$
```

Figure 28: Проверка снятия атрибута t с директории /tmp

Повтор предыдущих шагов

```
[guest2@smorodovadv guest]$ echo "test" > /tmp/file01.txt
[guest2@smorodovadv guest]$ echo "test2">> /tmp/file01.txt
[guest2@smorodovadv guest]$ cat /tmp/file01.txt
test
test2
[guest2@smorodovadv guest]$ echo "test3"> /tmp/file01.txt
[guest2@smorodovadv guest]$ cat /tmp/file01.txt
test3
[guest2@smorodovadv guest]$ rm /tmp/file01.txt
[guest2@smorodovadv guest]$
```

Figure 29: Повтор предыдущих шагов

Возвращение атрибута t директории /tmp

```
[guest2@smorodovadv guest]$ su -
Пароль:
[root@smorodovadv ~]# chmod +t /tmp
[root@smorodovadv ~]# exit
выход
[guest2@smorodovadv guest]$
```

Figure 30: Возвращение атрибута t директории /tmp

Выводы

Выводы

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