

Лабораторная работа №5

Информационная безопасность

Павлова П.А.

2024

Российский университет дружбы народов, Москва, Россия

- 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

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]

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]

Ход выполнения лабораторной работы

5.2.1. Подготовка лабораторного стенда

```
[paravloval2@paravloval2 ~]$ gcc -v
Используется внутренняя спецификация.
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 --enable-host-bfd-now --enable-languages=c,c++,fortran,lto --prefix=/usr --mandir=/usr/share/
man --infodir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-shared --enable-threads=posix --enable-checking=release --with-system-zlib --en
able__cxa_atexit --disable-libunwind-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-gcc-major-version-only --enable-plugin --enable-initfini-arr
ay --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-build-config=bootstrap-lto --enable-link-serialization=1
Модель многопоточности: posix
Supported LTO compression algorithms: zlib zstd
gcc версия 11.5.0 20240719 (Red Hat 11.5.0-2) (GCC)
[paravloval2@paravloval2 ~]$ setenforce 0
```

Рис. 1: (рис. 1. Установка gss)

5.3.1 Создание программы

Создали программу simpleid.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;
}
```

Рис. 2: (рис. 2. simpleid.c)

5.3.1 Создание программы

Скомпилировали и выполнили программу simpleid. Затем выполнили системную программу id и сравнили полученные результаты

```
[guest@papavlova12 ~]$ touch simpleid.c
[guest@papavlova12 ~]$ vi simpleid.c
[guest@papavlova12 ~]$ gcc simpleid.c -o simpleid
[guest@papavlova12 ~]$ ls
dir1  simpleid  simpleid.c
[guest@papavlova12 ~]$ ./simpleid
bash: ./simpleid: команда не найдена...
[guest@papavlova12 ~]$ ./simpleid
uid=1001, gid=1001
[guest@papavlova12 ~]$ id
uid=1001(guest) gid=1001(guest) rpyнны=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Рис. 3: (рис. 3. 3-5 пункты задания лабораторной)

5.3.1 Создание программы

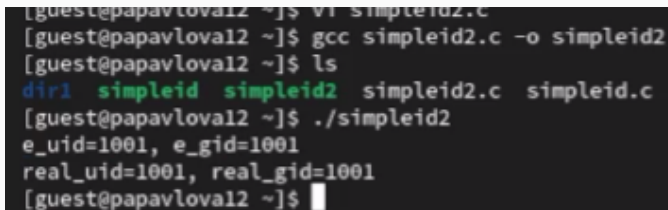
Усложнили программу, добавив вывод действительных идентификаторов

```
#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;
}
```

Рис. 4: (рис. 4. simpleid2.c)

5.3.1 Создание программы

Скомпилировали и выполнили программу simpleid2



```
[guest@papavlova12 ~]$ vi simpleid2.c
[guest@papavlova12 ~]$ gcc simpleid2.c -o simpleid2
[guest@papavlova12 ~]$ ls
dir1  simpleid  simpleid2  simpleid2.c  simpleid.c
[guest@papavlova12 ~]$ ./simpleid2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@papavlova12 ~]$
```

Рис. 5: (рис. 5. 7 пункт задания лабораторной)

5.3.1 Создание программы

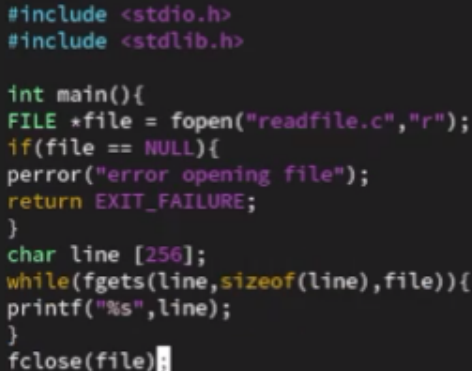
От имени суперпользователя выполнили команды и проверили правильность установки новых атрибутов и смены владельца файла. Запустили `simpleid2` и `id`. Сравнили результаты. Прodelали то же самое относительно SetGID-бита

```
[guest@papavloval2 ~]$ ls -l simpleid2
-rwsr-xr-x. 1 root guest 17656 сен  9 09:07 simpleid2
[guest@papavloval2 ~]$ ./simpleid2
e_uid=0, e_gid=1001
real_uid=1001, real_gid=1001
[guest@papavloval2 ~]$ id
uid=1001(guest) gid=1001(guest) rpynm=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@papavloval2 ~]$ touch readfile.c
[guest@papavloval2 ~]$ vi readfile.c
[guest@papavloval2 ~]$ gcc readfile.c -o readfile
[guest@papavloval2 ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе
```

Рис. 6: (рис. 6. 8-12 пункты задания лабораторной)

5.3.1 Создание программы

Скомпилировали программу readfile.c



```
#include <stdio.h>
#include <stdlib.h>

int main(){
    FILE *file = fopen("readfile.c","r");
    if(file == NULL){
        perror("error opening file");
        return EXIT_FAILURE;
    }
    char line [256];
    while(fgets(line,sizeof(line),file)){
        printf("%s",line);
    }
    fclose(file);
}
```

Рис. 7: (рис. 7. readfile.c)

5.3.1 Создание программы

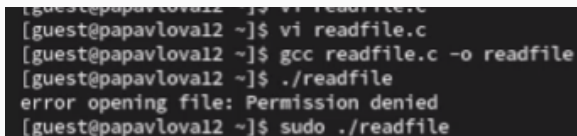
Сменили владельца у файла и изменили права так, чтобы только суперпользователь мог прочитать его, а guest не мог

```
[root@papavloval2 ~]# hown root:guest /home/guest/simpleid2
bash: hown: команда не найдена...
^[[[root@papavloval2 ~]# chown root:guest /home/guest/simpleid2
[root@papavloval2 ~]# chmod u+s /home/guest/simpleid2
[root@papavloval2 ~]# ls -l simpleid2
ls: невозможно получить доступ к 'simpleid2': Нет такого файла или каталога
[root@papavloval2 ~]# chown root:root /home/guest/readfile.c
[root@papavloval2 ~]# chmod 600 /home/guest/readfile.c
[root@papavloval2 ~]# chown root:root /home/guest/readfile
[root@papavloval2 ~]# chmod u+s /home/guest/readfile
[root@papavloval2 ~]# ./ /home/guest/readfile
-bash: ./: Это каталог
```

Рис. 8: (рис. 8. chmod)

5.3.1 Создание программы

Проверили, что guest не может прочитать файл. Сменили у программы readfile владельца и установили SetU'D-бит. Проверили, может ли программа readfile прочитать файл readfile.c, файл /etc/shadow



```
[guest@papavlova12 ~]$ vi readfile.c
[guest@papavlova12 ~]$ gcc readfile.c -o readfile
[guest@papavlova12 ~]$ ./readfile
error opening file: Permission denied
[guest@papavlova12 ~]$ sudo ./readfile
```

Рис. 9: (рис. 9. 16-19 пункты Guest)

5.3.1 Создание программы

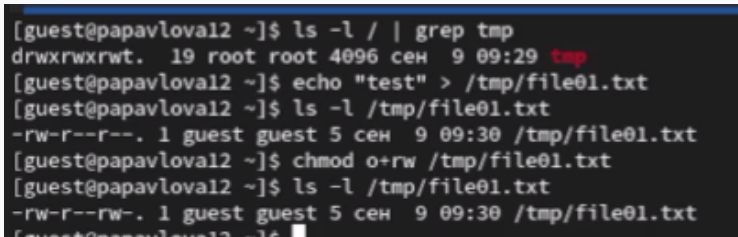
От имени суперпользователя все предыдущие команды удастся выполнить

```
[root@papavloval12 ~]# chown root:guest /home/guest/simpleid2
[root@papavloval12 ~]# chmod u+s /home/guest/simpleid2
[root@papavloval12 ~]# ls -l simpleid2
ls: невозможно получить доступ к 'simpleid2': Нет такого файла или каталога
[root@papavloval12 ~]# chown root:root /home/guest/readfile.c
[root@papavloval12 ~]# chmod 600 /home/guest/readfile.c
[root@papavloval12 ~]# chown root:root /home/guest/readfile
[root@papavloval12 ~]# chmod u+s /home/guest/readfile
[root@papavloval12 ~]# ./ /home/guest/readfile
-bash: ./: Это каталог
[root@papavloval12 ~]# cd /home/guest/readfile
-bash: cd: /home/guest/readfile: Это не каталог
[root@papavloval12 ~]# cd /home/guest/read
```

Рис. 10: (рис. 10. 16-18 пункты суперпользователь)

5.3.2. Исследование Sticky-бита

Выяснили, установлен ли атрибут Sticky на директории /tmp, создали файл file01.txt со словом test. Просмотрели атрибуты у только что созданного файла и разрешили чтение и запись для категории пользователей «все остальные»



```
[guest@papavloval2 ~]$ ls -l / | grep tmp
drwxrwxrwt. 19 root root 4096 сен  9 09:29 tmp
[guest@papavloval2 ~]$ echo "test" > /tmp/file01.txt
[guest@papavloval2 ~]$ ls -l /tmp/file01.txt
-rw-r--r--. 1 guest guest 5 сен  9 09:30 /tmp/file01.txt
[guest@papavloval2 ~]$ chmod o+rw /tmp/file01.txt
[guest@papavloval2 ~]$ ls -l /tmp/file01.txt
-rw-r--rw-. 1 guest guest 5 сен  9 09:30 /tmp/file01.txt
[guest@papavloval2 ~]$
```

Рис. 11: (рис. 12. 1-3 пункты)

5.3.2. Исследование Sticky-бита

От guest2 попробовали прочесть файл, дозаписать слово test2, затем записать слово test3, стерев при этом всю имеющуюся в файле информацию. Попробовали удалить файл. Этого сделать не удалось.

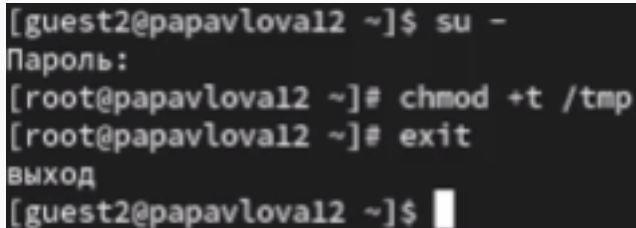
Повысили свои права до суперпользователя и сняли атрибут t с директории /tmp. От guest2 проверили, что атрибута t у директории /tmp нет

```
Пароль:
[guest2@papavloval2 ~]$ cat /tmp/file01.txt
test
[guest2@papavloval2 ~]$ echo "test2" >> /tmp/file01.txt
-bash: /tmp/file01.txt: Отказано в доступе
[guest2@papavloval2 ~]$ echo "test2" > /tmp/file01.txt
-bash: /tmp/file01.txt: Отказано в доступе
[guest2@papavloval2 ~]$ cat /tmp/file01.txt
test
[guest2@papavloval2 ~]$ rm /tmp/file01.txt
rm: удалить защищённый от записи обычный файл '/tmp/file01.txt'? y
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
[guest2@papavloval2 ~]$ su -
Пароль:
[root@papavloval2 ~]# chmod -t /tmp
[root@papavloval2 ~]# exit
```

5.3.2. Исследование Sticky-бита

Повторили предыдущие шаги. При повторении всё получилось. Удалось удалить файл от имени пользователя, не являющегося его владельцем.

Повысили свои права до суперпользователя и вернули атрибут `t` на директорию `/tmp`



```
[guest2@papavlova12 ~]$ su -  
Пароль:  
[root@papavlova12 ~]# chmod +t /tmp  
[root@papavlova12 ~]# exit  
выход  
[guest2@papavlova12 ~]$
```

Рис. 13: (рис. 15. Возвращение атрибута)

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

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>