

Отчет по лабораторной работе №5

Основы информационной безопасности

Машков Илья Евгеньевич

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	16
	Список литературы	17

Список иллюстраций

2.1	Подготовка к лабораторной работе	6
2.2	Создание файла	6
2.3	Содержимое файла	7
2.4	Компиляция файла	7
2.5	Сравнение команд	7
2.6	Создание и компиляция файла	8
2.7	Содержимое файла	8
2.8	Смена владельца файла и прав доступа к файлу	8
2.9	Запуск файла	9
2.10	Создание и компиляция файла	9
2.11	Содержимое файла	10
2.12	Смена владельца файла и прав доступа к файлу	10
2.13	Попытка прочесть содержимое файла	10
2.14	Попытка прочесть содержимое файла программой	11
2.15	Попытка прочесть содержимое файла программой	11
2.16	Чтение файла от имени суперпользователя	11
2.17	Проверка атрибутов директории tmp	11
2.18	Создание файла, изменение прав доступа	12
2.19	Попытка чтения файла	12
2.20	Попытка записи в файл	12
2.21	Попытка удалить файл	12
2.22	Смена атрибутов файла	13
2.23	Проверка атрибутов директории	13
2.24	Повтор предыдущих действий	14
2.25	Изменение атрибутов	15

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 Выполнение лабораторной работы

Для лабораторной работы необходимо проверить, установлен ли компилятор gcc, команда `gcc -v` позволяет это сделать. Также осуществляется отключение системы запретов с помощью `setenforce 0` (рис. 1).

```
[guest@localhost ~]$ su iemashkov
Пароль:
[iemashkov@localhost guest]$ whereis gcc
gcc: /usr/bin/gcc /usr/lib/gcc /usr/libexec/gcc /usr/share/man/man1/gcc.1.gz /usr/share/info/gcc.info.gz
[iemashkov@localhost guest]$ whereis g++
g++: /usr/bin/g++ /usr/share/man/man1/g++.1.gz
[iemashkov@localhost guest]$ gcc -v
Используются внутренние спецификации.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/8/lto-wrapper
OFFLOAD_TARGET_NAMES=nvptx-none
OFFLOAD_TARGET_DEFAULT=1
Целевая архитектура: x86_64-redhat-linux
Параметры конфигурации: ../configure --enable-bootstrap --enable-languages=c,c++,fortran,lto --prefix=/usr
--mandir=/usr/share/man --infodir=/usr/share/info --with-bugurl=https://bugs.rockylinux.org/ --enable-s
hared --enable-threads=posix --enable-checking=release --enable-multilib --with-system-zlib --enable-__cx
a_atexit --disable-libunwind-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-gcc-ma
jor-version-only --with-linker-hash-style=gnu --enable-plugin --enable-initfini-array --with-isl --disabl
e-libmpx --enable-offload-targets=nvptx-none --without-cuda-driver --enable-gnu-indirect-function --enabl
e-cet --with-tune=generic --with-arch_32=x86-64 --build=x86_64-redhat-linux
Модель многопоточности: posix
gcc версия 8.5.0 20210514 (Red Hat 8.5.0-22) (GCC)
[iemashkov@localhost guest]$ sudo setenforce 0
[sudo] пароль для iemashkov:
[iemashkov@localhost guest]$ getenforce
Permissive
[iemashkov@localhost guest]$
```

Рис. 2.1: Подготовка к лабораторной работе

Осуществляю вход от имени пользователя `guest` и создаю файл `simplified.c` и записываю в файл код (рис. 2)

```
[guest@localhost ~]$ touch simplified.c
[guest@localhost ~]$ nano simplified.c
[guest@localhost ~]$
```

Рис. 2.2: Создание файла

```
C++ Листинг 1 #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; }
```

Содержимое файла выглядит следующим образом (рис. 3)

```
GNU nano 2.9.8

#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. 2.3: Содержимое файла

Компилирую файл, проверяю, что он скомпилировался (рис. 4)

```
[guest@localhost ~]$ gcc simplified.c -o simplified
[guest@localhost ~]$ ls
dirl    simplified.c  Документы  Изображения  Общедоступные  Шаблоны
simplified  Видео      Загрузки  Музыка      'Рабочий стол'
```

Рис. 2.4: Компиляция файла

Запускаю исполняемый файл. В выводе файла выписаны номера пользователя и групп, от вывода при вводе `if`, они отличаются только тем, что информации меньше (рис. 5)

```
[guest@localhost ~]$ ./simplified
uid=1001, gid=1001
[guest@localhost ~]$ ig
bash: ig: команда не найдена...
[guest@localhost ~]$ id
uid=1001(guest) gid=1001(guest) rpnpm=1001(guest) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[guest@localhost ~]$
```

Рис. 2.5: Сравнение команд

Создание, запись в файл и компиляция файла `simplified2.c`. Запуск программы (рис. 6)

```
[guest@localhost ~]$ gcc simplified2.c -o simplified2
[guest@localhost ~]$ ./simplified2
e_uid=1001, e_gid=1001
real_uid=1001, real_gid=1001
[guest@localhost ~]$
```

Рис. 2.6: Создание и компиляция файла

C++ Листинг 2 `#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. 7)

```
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
int
main()
{
uid_t real_uid = geteuid();
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. 2.7: Содержимое файла

С помощью `chown` изменяю владельца файла на суперпользователя, с помощью `chmod` изменяю права доступа (рис. 8)

```
[iemashkov@localhost guest]$ sudo chown root:guest /home/guest/simplified2
[sudo] пароль для iemashkov:
[iemashkov@localhost guest]$ sudo chmod u+s /home/guest/simplified2
[iemashkov@localhost guest]$ sudo ls -l /home/guest/simplified2
-rwsrwxr-x. 1 root guest 18208 anp 19 20:25 /home/guest/simplified2
[iemashkov@localhost guest]$
```

Рис. 2.8: Смена владельца файла и прав доступа к файлу

Сравнение вывода программы и команды `id`, наша команда снова вывела только ограниченное количество информации(рис. 9)


```

[iemashkov@localhost guest]$ sudo /home/guest/simplified2
e_uid=0, e_gid=0
real_uid=0, real_gid=0
[iemashkov@localhost guest]$ id
uid=1000(iemashkov) gid=1000(iemashkov) rpyнны=1000(iemashkov),10(wheel) контекст=unconfined_u:unconfined
_r:unconfined_t:s0-s0:c0.c1023
[iemashkov@localhost guest]$ sudo id
uid=0(root) gid=0(root) rpyнны=0(root) контекст=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[iemashkov@localhost guest]$

```

Рис. 2.9: Запуск файла

Создание и компиляция файла readfile.c (рис. 10)

```

[guest@localhost ~]$ touch readfile.c
[guest@localhost ~]$ nano readfile.c
[guest@localhost ~]$ gcc readfile.c -o readfile
[guest@localhost ~]$ ls
dir1      simplified      simplified.c    Загрузки      Общедоступные
readfile  simplified2     Видео          Изображения   'Рабочий стол'
readfile.c  simplified2.c  Документы     Музыка        Шаблоны
[guest@localhost ~]$

```

Рис. 2.10: Создание и компиляция файла

C++ Листинг 3

```

#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; }

```

(рис. 11)

```

#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;
}

```

Рис. 2.11: Содержимое файла

Снова от имени суперпользователя меняю владельца файла readfile. Далее меняю права доступа так, чтобы пользователь guest не смог прочесть содержимое файла (рис. 12)

```

[iemashkov@localhost guest]$ sudo chown root:guest /home/guest/readfile.c
[sudo] пароль для iemashkov:
[iemashkov@localhost guest]$ sudo chmod u+s /home/guest/readfile.c
[iemashkov@localhost guest]$ sudo chmod 700 /home/guest/readfile.c
[iemashkov@localhost guest]$ sudo chmod -r /home/guest/readfile.c
[iemashkov@localhost guest]$ sudo chmod u+s /home/guest/readfile.c
[iemashkov@localhost guest]$

```

Рис. 2.12: Смена владельца файла и прав доступа к файлу

Проверка прочесть файл от имени пользователя guest. Прочесть файл не удастся (рис. 13)

```

[guest@localhost ~]$ cat readfile.c
cat: readfile.c: Отказано в доступе

```

Рис. 2.13: Попытка прочесть содержимое файла

Попытка прочесть тот же файл с помощью программы readfile, в ответ получаем “отказано в доступе” (рис. 14)



Рис. 2.14: Попытка прочесть содержимое файла программой

Попытка прочесть файл `\etc\shadow` с помощью программы, все еще получаем отказ в доступе (рис. 15)



Рис. 2.15: Попытка прочесть содержимое файла программой

Пробуем прочесть эти же файлы от имени суперпользователя и чтение файлов проходит успешно (рис. 16)



Рис. 2.16: Чтение файла от имени суперпользователя

Проверяем папку tmp на наличие атрибута Sticky, т.к. в выводе есть буква t, то атрибут установлен (рис. 17)

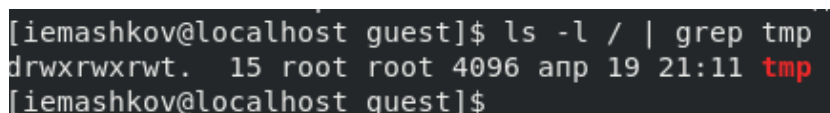


Рис. 2.17: Проверка атрибутов директории tmp

От имени пользователя guest создаю файл с текстом, добавляю права на чтение и запись для других пользователей (рис. 18)

```
[guest@localhost ~]$ echo "test" > /tmp/file01.txt
[guest@localhost ~]$ ls -l /tmp/file01.txt
-rw-rw-r--. 1 guest guest 5 anp 19 21:15 /tmp/file01.txt
[guest@localhost ~]$ chmod o+rw /tmp/file01.txt
[guest@localhost ~]$ ls -l /tmp/file01.txt
-rw-rw-rw-. 1 guest guest 5 anp 19 21:15 /tmp/file01.txt
[guest@localhost ~]$
```

Рис. 2.18: Создание файла, изменение прав доступа

Вхожу в систему от имени пользователя guest2, от его имени могу и прочитать, и дозаписать файл file01.txt (рис. 19)

```
[guest@localhost ~]$ su guest2
Пароль:
[guest2@localhost guest]$ cat /tmp/file01.txt
test
[guest2@localhost guest]$ echo "test2" >> /tmp/file01.txt
[guest2@localhost guest]$ cat /tmp/file01.txt
test
test2
[guest2@localhost guest]$
```

Рис. 2.19: Попытка чтения файла

Также могу добавить в файл file01.txt новую информацию от имени пользователя guest2 (рис. 20)

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

Рис. 2.20: Попытка записи в файл

Далее пробуем удалить файл, и получаем отказ (рис. 21)

```
[guest2@localhost guest]$ rm /tmp/file01.txt
rm: невозможно удалить '/tmp/file01.txt': Операция не позволена
[guest2@localhost guest]$
```

Рис. 2.21: Попытка удалить файл

От имени суперпользователя снимаем с директории атрибут Sticky (рис. 22)

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

Рис. 2.22: Смена атрибутов файла

Проверяем, что атрибут действительно снят (рис. 23)

```
[guest2@localhost guest]$ ls -l / | grep tmp  
drwxrwxrwx. 15 root root 4096 апр 19 21:32 tmp  
[guest2@localhost guest]$
```

Рис. 2.23: Проверка атрибутов директории

Далее был выполнен повтор предыдущих действий. По результатам без Sticky-бита удаление файла, запись и дозапись в файл осталась возможной (рис. 24)

```

[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ echo "test2" >> /tmp/file01.txt
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
test2
[guest2@localhost guest]$ echo "test3" > /tmp/file01.txt
[guest2@localhost guest]$ cat /tmp/file01.txt
test3
[guest2@localhost guest]$ rm /tmp/file01.txt
[guest2@localhost guest]$ ls -l /| grep tmp
drwxrwxrwx. 15 root root 4096 апр 19 21:36 tmp
[guest2@localhost guest]$ ls -l
итого 72
drwxrwxr-x. 2 guest guest 19 апр 5 15:34 dir1
-rwxrwxr-x. 1 guest guest 18256 апр 19 20:42 readfile
--ws----- 1 root guest 402 апр 19 20:41 readfile.c
-rwxrwxr-x. 1 guest guest 18208 апр 19 20:14 simplified
-rwsrwxr-x. 1 root guest 18208 апр 19 20:25 simplified2
-rw-rw-r-- 1 guest guest 299 апр 19 20:24 simplified2.c
-rw-rw-r-- 1 guest guest 171 апр 19 20:13 simplified.c
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Видео
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Документы
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Загрузки
drwxr-xr-x. 2 guest guest 192 мар 8 19:34 Изображения
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Музыка
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Общедоступные
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 'Рабочий стол'
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Шаблоны
[guest2@localhost guest]$ ls -l /home/guest
итого 72
drwxrwxr-x. 2 guest guest 19 апр 5 15:34 dir1
-rwxrwxr-x. 1 guest guest 18256 апр 19 20:42 readfile
--ws----- 1 root guest 402 апр 19 20:41 readfile.c
-rwxrwxr-x. 1 guest guest 18208 апр 19 20:14 simplified
-rwsrwxr-x. 1 root guest 18208 апр 19 20:25 simplified2
-rw-rw-r-- 1 guest guest 299 апр 19 20:24 simplified2.c
-rw-rw-r-- 1 guest guest 171 апр 19 20:13 simplified.c
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Видео
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Документы
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Загрузки
drwxr-xr-x. 2 guest guest 192 мар 8 19:34 Изображения
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Музыка
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Общедоступные
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 'Рабочий стол'
drwxr-xr-x. 2 guest guest 6 мар 8 19:25 Шаблоны

```

Рис. 2.24: Повтор предыдущих действий

Возвращение директории tmp атрибута t от имени суперпользователя (рис. 25)

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

Рис. 2.25: Изменение атрибутов

3 Выводы

Изучил механизм изменения идентификаторов, применила SetUID- и Sticky-биты. Получила практические навыки работы в консоли с дополнительными атрибутами. Рассмотрела работы механизма смены идентификатора процессов пользователей, а также влияние бита Sticky на запись и удаление файлов.

Список литературы

Основы информационной безопасности