

Презентация №5

Дискреционное разграничение прав в Linux. Исследование влияния дополнительных атрибутов

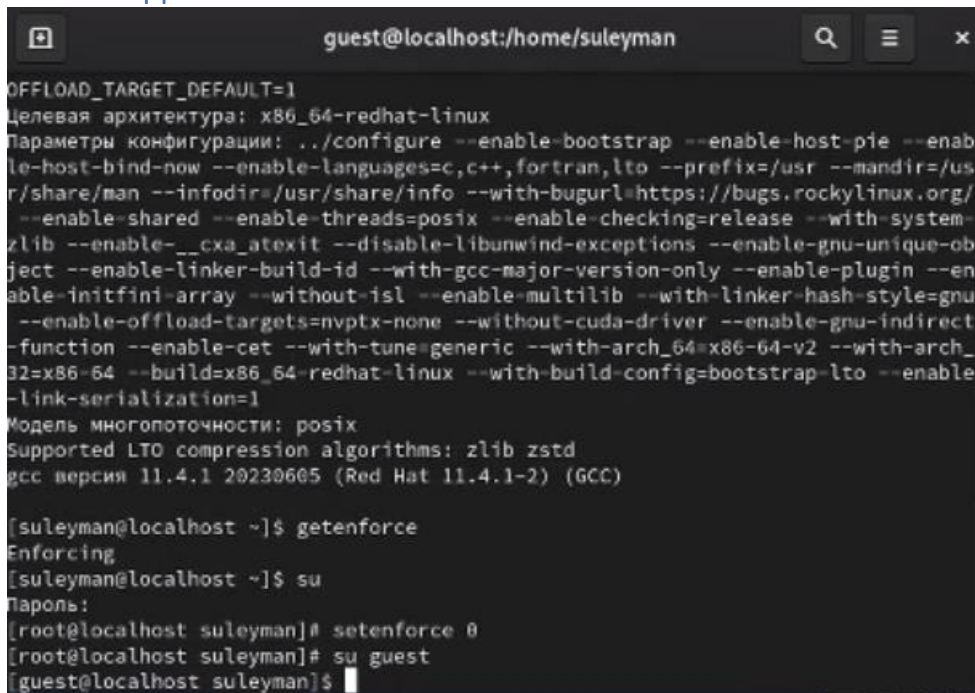
Эттеев Сулейман

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 работы

2.1 Подготовка



```
guest@localhost:/home/suleyman
OFFLOAD_TARGET_DEFAULT=1
Целевая архитектура: x86_64-redhat-linux
Параметры конфигурации: ../configure --enable-bootstrap --enable-host-pie --enable-host-bind-now --enable-languages=c,c++,fortran,lto --prefix=/usr --mandir=/usr/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-object --enable-linker-build-id --with-gcc-major-version-only --enable-plugin --enable-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-build-config=bootstrap-lto --enable-link-serialization=1
Модель многопоточности: posix
Supported LTO compression algorithms: zlib zstd
gcc версия 11.4.1 20230605 (Red Hat 11.4.1-2) (GCC)

[suleyman@localhost ~]$ getenforce
Enforcing
[suleyman@localhost ~]$ su
Пароль:
[root@localhost suleyman]# setenforce 0
[root@localhost suleyman]# su guest
[guest@localhost suleyman]$
```

Figure 1: подготовка к работе

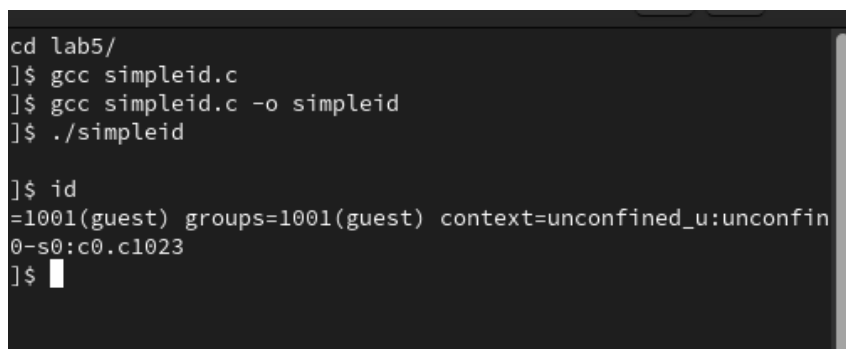
2.2 Изучение механики SetUID



```
simpleid.c
~/lab5

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

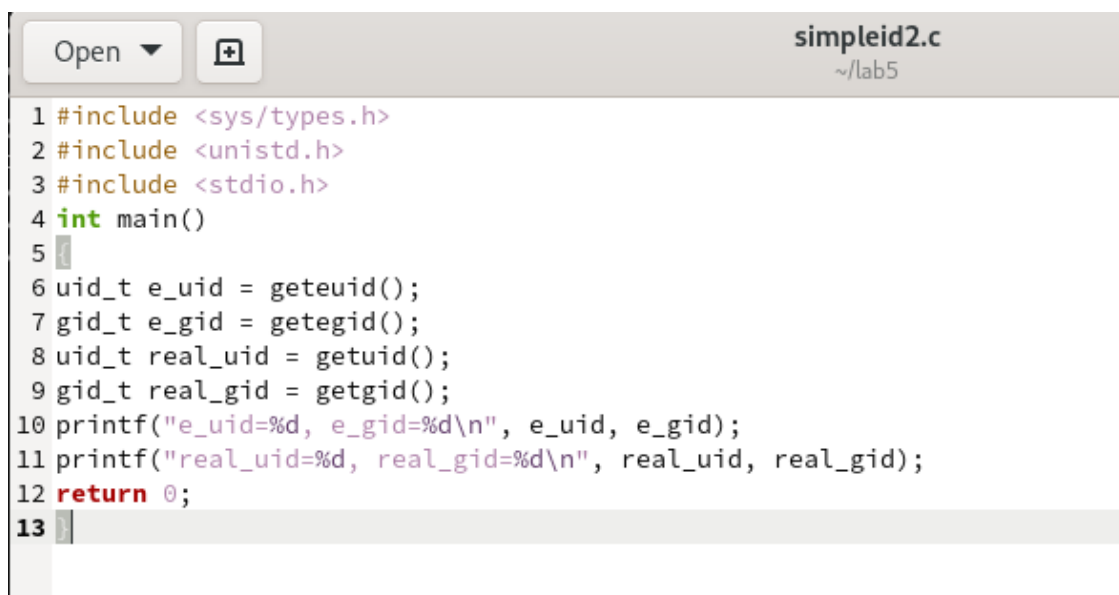
Figure 2: программа simpleid



```
cd lab5/
]$ gcc simpleid.c
]$ gcc simpleid.c -o simpleid
]$ ./simpleid

]$ id
=1001(guest) groups=1001(guest) context=unconfined_u:unconfined_u:s0:c0.c1023
]$
```

Figure 3: результат программы simpleid



```
simpleid2.c
~/lab5

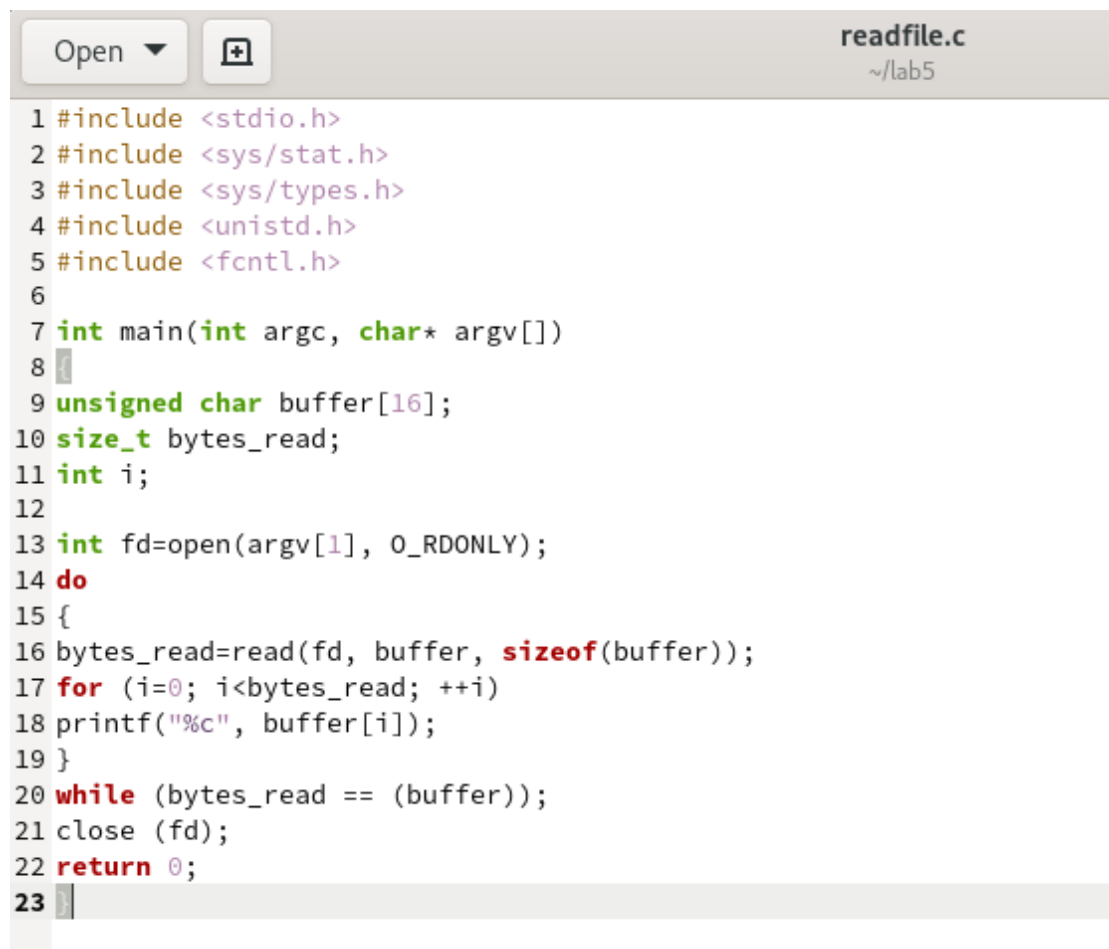
1 #include <sys/types.h>
2 #include <unistd.h>
3 #include <stdio.h>
4 int main()
5 {
6     uid_t e_uid = geteuid();
7     gid_t e_gid = getegid();
8     uid_t real_uid = getuid();
9     gid_t real_gid = getgid();
10    printf("e_uid=%d, e_gid=%d\n", e_uid, e_gid);
11    printf("real_uid=%d, real_gid=%d\n",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

Figure 4: программа simpleid2

12. Проделали тоже самое относительно SetGID-бита.

```
[suleyman@localhost ~]$ su guest
Пароль:
[guest@localhost suleyman]$ ./simpleid.c
bash: ./simpleid.c: Отказано в доступе
[guest@localhost suleyman]$ su
Пароль:
[root@localhost suleyman]# ./simpleid
bash: ./simpleid: Отказано в доступе
[root@localhost suleyman]# id
uid=0(root) gid=0(root) группы=0(root) контекст=unconfined_u:unconfined_r:unconf
ined_t:s0-s0:c0.c1023
[root@localhost suleyman]#
```

Figure 5: результат программы simpleid2



```
readfile.c
~/lab5

1 #include <stdio.h>
2 #include <sys/stat.h>
3 #include <sys/types.h>
4 #include <unistd.h>
5 #include <fcntl.h>
6
7 int main(int argc, char* argv[])
8 {
9     unsigned char buffer[16];
10    size_t bytes_read;
11    int i;
12
13    int fd=open(argv[1], O_RDONLY);
14    do
15    {
16    bytes_read=read(fd, buffer, sizeof(buffer));
17    for (i=0; i<bytes_read; ++i)
18    printf("%c", buffer[i]);
19    }
20    while (bytes_read == (buffer));
21    close (fd);
22    return 0;
23 }
```

Figure 6: программа readfile

Figure 7: результат программы readfile

2.3 Исследование Sticky-бита

```
$ echo test >> /tmp/file01.txt
$ chmod g+rwX /tmp/file01.txt
$ su guest2

[]$ cd /tmp/
$ cat file01.txt

$ echo test2 >> file01.txt
$ cat file01.txt

$ rm file01.txt
file01.txt': Operation not permitted
$ su

chmod -t /tmp

$ echo test 2 >> file01.txt
$ rm file01.txt
$ su

chmod +t /tmp

$
$
```

Figure 8: исследование Sticky-бита

3 Выводы

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

Список литературы

1. [КОМАНДА CHATTR В LINUX](#)
2. [chattr](#)