# Programming Assignment № 6

潘亦晟, 515021910384

04/16/2018

# **Problem 1**

问题: 需要使非管理员用户可以读取但是无法改写或者删除任何文件

## 难点:

- 普通用户无法读取某些/root 目录下的文件。
- 使用 root 权限 chmod u+s 命令会为文件设置 SetUID, 使得普通用户获得读和写的 root 权限。

#### **Definition:**

- UID: 用来标识系统中各个不同的用户。
- ·eUID: 运行程序时的有效身份。

我们可以通过 setuid() 和 seteuid() 函数设置 UID 和 eUID。 我们可以通过 getuid() 和 geteuid() 函数读取 UID 和 eUID。

思路: 我们仅在需要在执行高权限读取前将 eUID 设为 root 用户 UID, 在读取后将 eUID 更改回为普通用户 UID, 实现类似针对 read 操作的 chmod。

## 寻找 root 和普通用户 UID

1 pys666@ubuntu:~\$ cat \etc\passwd

我们发现: uid\_root = 0 ,uid\_user = 1000。

#### 代码实现

#### Listing 1: read all the linux file

- 1 //一个用于所有用户读取Unix系统中所有文件的程序, 使用方式: 程序名 文件名。
- 2 #include<stdio.h>
- 3 #include<string.h>
- 4 #include<stdlib.h>
- 5 #include<sys/types.h>

Assignment № 6 Page 1

```
6 #include<fcntl.h>
7 int main (int argc, char *argv[])
8 {
9
10 //降低权限
11 seteuid(1000);
12
13 FILE *fp;
14 int ch;
15
16 //对于高权限读取降权
17 if(access(argv[1],R_OK)!=0){
18 seteuid(0);
19 }
20
21 fp = fopen(argv[1], "r");
22
23 //得到文件指针后立即降低权限
24 seteuid(1000);
25
26 if (fp == NULL)
27 {
28 printf("open file %s failed", argv[1]);
29 }
30 else
31 {
32 printf("open file %s successfully", argv[1]);
33 }
34
35
36
37 fclose(fp);
38
39 return 0;
40 }
```

# 测试文件

- 创建一个 text 文件,使用 chmod 700 text 命令更改文件权限,设置拥有者可读写执行,其他人不可读写执行。
- 使用 Is -I text 命令查看文件权限。

Assignment № 6 Page 2

```
pys666@ubuntu:~/Downloads$ sudo chmod 700 text
pys666@ubuntu:~/Downloads$ ls -l text
-rwx----- 1 root root 9 Apr 15 20:25 text
```

图 1:

## 实验结果

我们使用以下命令编译源代码:

- sudo gcc superread.c -o superread
- sudo chmod u+s superread
- ./superread text

```
pys666@ubuntu:~/Downloads$ sudo chmod u+s superread
pys666@ubuntu:~/Downloads$ ./superread text
open file text successfullypys666@ubuntu:~/Downloads$
```

图 2: result

读取 text 文件成功,功能实现。

Assignment № 6 Page 3