[pwnable.kr WriteUp]

이태규(ltg2030)

[Toddler's Bottle] random 1PT

문제 설명
Daddy, teach me how to use random value in programming! ssh random@pwnable.kr -p2222 (pw:guest)

rand 함수 : 씨앗 (seed값)을 이용하여 랜덤처럼 보이는 값을 생성해주는 함수 (컴퓨터로 이론적으로 완전한 랜덤 값은 만들 수 없기 때문에 seed 값을 통해 다음 값을 구해낸다.)

이 문제에서 seed값의 변화를 주지 않기 때문에 처음 생성되는 랜덤값이 고정된 값이다. 따라서 key에 그 생성된 랜덤 값과 0xdeadbeef를 xor한 값으로 넣어주게 되면 key와 생성된 랜덤 값을 xor 하면 0xdeadbeef가 나오게 된다.

문제 풀이(이론)

1. 우선 Xshell을 통해 random@pwnable.kr -p2222에 접속한 뒤 ls -l 명령어를 통해 어떤 파일들이 있는지 확인했다.

2.
```
-r--r----- 1 random2 root      49 Jun 30  2014 flag
-r-sr-x--- 1 random2 random 8538 Jun 30  2014 random
-rw-r--r-- 1 root    root     301 Jun 30  2014 random.c
```
이런 3가지 파일이 있는 것을 알 수 있다. 우리는 지금 random라는 사용자를 사용하고 있고 이는 root 그룹이 아니기 때문에 flag를 직접 열어볼 수는 없고 SetUID가 적용된 random 프로그램을 이용하여 이를 열어야 하는 것을 추측할 수 있다.

3. random.c를 열어 random 프로그램이 어떻게 동작하는지 알아보자. 프로그램에서 랜덤 값과 xor하여 0xdeadbeef와 같은지 확인하여 flag를 출력해 주는데 위에서 얘기했드시 생성된 랜덤 값이 고정적이기 때문에 그 값과 0xdeadbeef를 xor한 값을 구해낼 수 있고 그것을 key로서 입력하면 조건문을 만족시킬 수 있다.

4. 직접 리눅스에서 seed값의 초기화 없이 처음 생성되는 랜덤값을 구해보면 1804289383가 되고 이를 0xdeadbeef와 xor한 값은 -1255736440이 되고 이를 입력해주면 flag를 얻을 수 있다.

문제 풀이(실습)

[pwnable.kr WriteUp]

이태규(ltg2030)

```
random@ubuntu:~$ ls -l
total 20
-r--r----- 1 random2 root     49 Jun 30  2014 flag
-r-sr-x--- 1 random2 random 8538 Jun 30  2014 random
-rw-r--r-- 1 root    root    301 Jun 30  2014 random.c
random@ubuntu:~$ cat random.c
#include <stdio.h>

int main(){
        unsigned int random;
        random = rand();        // random value!

        unsigned int key=0;
        scanf("%d", &key);

        if( (key ^ random) == 0xdeadbeef ){
                printf("Good!\n");
                system("/bin/cat flag");
                return 0;
        }

        printf("Wrong, maybe you should try 2^32 cases.\n");
        return 0;
}

random@ubuntu:~$ ./random
-1255736440
Good!
Mommy, I thought libc random is unpredictable...
```

good job :)\n 이 출력된 뒤 flag를 cat 하기 때문에 mommy! I think I know what a file descriptor is!! 가 flag 내용임을 알 수 있다.

실행화면