[pwnable.kr WriteUp]

이태규(ltg2030)

[Toddler's Bottle] collision 3PT

문제 설명
Daddy told me about cool MD5 hash collision today. I wanna do something like that too! ssh col@pwnable.kr -p2222 (pw:guest)

MD5 hash : 128비트 암호화 해시 함수로 무결성 검사 등에 사용된다.
Collision : 어떤 해시 함수 H에 대해 $X \neq Y$며 $H(X) = H(Y)$ 인 경우 충돌이 일어났다고 하고 X, Y를 충돌 쌍이라고 한다.

리눅스 기본
리눅스에서 파라미터로 프로그램에 ascii코드가 키보드로 입력할 수 없는 즉 0x01, 0xEC 이런 수를 입력하기 위해 python 코드를 사용할 수 있다.
(ex ./col `python -c 'print ("\xEC\x01")'`
※ 주의 python 명령어 앞에 붙는 기호는 따옴표가 아닌 보통 키보드의 1왼쪽에 있는 `이다. 이것 하나 때문에 약 2시간 정도 문제 풀이가 지연됐는데 항상 주의하자!

문제 풀이(이론)

[pwnable.kr WriteUp]

이태규(ltg2030)

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

2.
-r-sr-x--- 1 col2 col 7341 Jun 11 2014 col
-rw-r--r-- 1 root root 555 Jun 12 2014 col.c
-r--r----- 1 col2 col2 52 Jun 11 2014 flag
이런 3가지 파일이 있는 것을 알 수 있다. 우리는 지금 fd라는 사용자를 사용하고 있고 이는 root 그룹이 아니기 때문에 flag를 직접 열어볼 수는 없고 SetUID가 적용된 col 프로그램을 이용하여 이를 열어야 하는 것을 추측할 수 있다.

3. col.c를 열어 col 프로그램이 어떻게 동작하는지 알아보자. cat col.c 명령어를 사용하여 col.c의 내용을 확인한 결과 shell에서 파라미터로 문자열을 입력받고 그 문자열이 20글자가 되지 않는 경우는 제외시킨다. 만약 문자열이 20글자인 경우 4개 단위로 끊어 5개의 int형으로 합친 뒤 그것을 더해 hash code와 같아지도록 하는 것이 목표이다.

4. 어떤 문자열 'abcd'를 int형으로 만들면 little endian이 적용되어 'd'*2^24 + 'c'*2^16 + 'b'*2 ^ 8 + 'a'로 바뀌기 때문에 이를 생각하여 문자열을 입력해주면 된다.

5. hash code = 0x21DD09EC = 0x01010101 * 4 + 0x1dd905e8 로 생각할 수 있기 때문에 문자열을 \x01을 16번 넣어주고 \xe8\x05\xd9\x1d 순으로 넣어주면 된다.

문제 풀이(실습)

```
col@ubuntu:~$ ls -l
total 16
-r-sr-x--- 1 col2 col  7341 Jun 11  2014 col
-rw-r--r-- 1 root root  555 Jun 12  2014 col.c
-r--r----- 1 col2 col2   52 Jun 11  2014 flag
col@ubuntu:~$ cat col.c
#include <stdio.h>
#include <string.h>
unsigned long hashcode = 0x21DD09EC;
unsigned long check_password(const char* p){
        int* ip = (int*)p;
        int i;
        int res=0;
        for(i=0; i<5; i++){
                res += ip[i];
        }
        return res;
}

int main(int argc, char* argv[]){
        if(argc<2){
                printf("usage : %s [passcode]\n", argv[0]);
                return 0;
        }
        if(strlen(argv[1]) != 20){
                printf("passcode length should be 20 bytes\n");
                return 0;
        }

        if(hashcode == check_password( argv[1] )){
                system("/bin/cat flag");
                return 0;
        }
        else
                printf("wrong passcode.\n");
        return 0;
}
col@ubuntu:~$ ./col `python -c 'print ("\x01"*16+"\xe8\x05\xd9\x1d")'`
daddy! I just managed to create a hash collision :)
```

입력을 넣어주고 나온 daddy! I just managed to create a hash collision :) 이 flag임을 알 수 있다.

실행화면