```
passcode@ubuntu:~$ file passcode
passcode: setgid ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically link
ed, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.24, BuildID[sha1]=d2b7bd64f70e46b1b0e
b7036b35b24a651c3666b, not stripped
passcode@ubuntu:~$ checksec passcode
[*] '/home/passcode/passcode'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      No PIE
passcode@ubuntu:~$ exit
logout
Connection to pwnable.kr closed.
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr  ssh passcode@pwnable.kr -p2222
passcode@pwnable.kr's password:
```

Ret's pwn~!
pwnable.kr write up

2018-07-05
황선홍(f.killrra)

```
- Site admin : daehee87.kr@gmail.com
- IRC : irc.netgarage.org:6667 / #pwnable.kr
- Simply type "irssi" command to join IRC now
- files under /tmp can be erased anytime. make your directory under /tmp
- to use peda, issue `source /usr/share/peda/peda.py` in gdb terminal
Last login: Wed Jul  4 00:11:53 2018 from 176.12.222.219
passcode@ubuntu:~$
```

[목차]

- 0x01 fd
  - File descriptor에 대해 알아보자.

- 0x02 random
  - rand()의 취약점에 대해 알아보자.

[Toddler's Bottle] 0x01

0x01 fd
-> What is a file descriptor in Linux?

[file descriptor]
- 시스템으로 부터 할당 받은 파일을 대표하는 0이 아닌 정수 값
- 프로세스에서 열린 파일의 목록을 관리하는 테이블의 인덱스

Integer value	Name	<unistd.h> symbolic constant[1]	<stdio.h> file stream[2]
0	Standard input	STDIN_FILENO	stdin
1	Standard output	STDOUT_FILENO	stdout
2	Standard error	STDERR_FILENO	stderr

[fd.c]

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char buf[32];
int main(int argc, char* argv[], char* envp[]){
    if(argc<2){
        printf("pass argv[1] a number\n");
        return 0;
    }
    int fd = atoi( argv[1] ) - 0x1234;
    int len = 0;
    len = read(fd, buf, 32);
    if(!strcmp("LETMEWIN\n", buf)){
        printf("good job :)\n");
        system("/bin/cat flag");
        exit(0);
    }
    printf("learn about Linux file IO\n");
    return 0;

}

```

1. argc 즉 인자가 파일명 빼고 하나 더 있어야 합니다.

2. fd라는 변수에 atoi()로 argv[1]을 받아 정수형으로 반환하고 0x1234를 뺀 뒤 저장합니다.

3. len 변수에 read()의 return 값을 저장합니다.

4. read()에서 fd가 바라보고 있는 파일에서 32byte길이 만큼 buf에 읽어들입니다.

5. strcmp()로 buf에서 LETMEWIN₩n이라는 값을 비교하고 맞다면 flag값을 열어줍니다.

*간접적인 풀이만 하겠습니다.

자자. 정답은 집가서 생각해보시고....

*간접적인 풀이만 하겠습니다.

0x02 random

-> Daddy, teach me how to use random value in programming!

[rand()]
- 랜덤한 값을 뱉는 함수
- 취약점이 있는데 그게 말이야.....

test.c (/mnt/hgfs/vm_shared/pwnable.kr/tmp) - VIM

```c
#include <stdio.h>

int main()
{
    unsigned int random;
    random = rand();

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

    printf("key ^ random = %x\n", key ^ random);
    printf("key = %x\n", key);
    printf("random = %x\n", random);
    return 0;
}
```

```
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp ./a.out
0x1234
key ^ random = 6b8b4567
key = 0
random = 6b8b4567
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp ./a.out
0x1234
key ^ random = 6b8b4567
key = 0
random = 6b8b4567
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp ./a.out
1234
key ^ random = 6b8b41b5
key = 4d2
random = 6b8b4567
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp ./a.out
1234
key ^ random = 6b8b41b5
key = 4d2
random = 6b8b4567
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp ./a.out
2345
key ^ random = 6b8b4c4e
key = 929
random = 6b8b4567
fkillrra@ubuntu /mnt/hgfs/vm_shared/pwnable.kr/tmp
```

....?

test.c (/mnt/hgfs/vm_shared/pwnable.kr/tmp) - VIM

```c
#include <stdio.h>

int main()
{
    unsigned int random;
    random = rand();

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

    printf("key ^ random = %x\n", key ^ random);
    printf("key = %x\n", key);
    printf("random = %x\n", random);
    return 0;
}
```


```
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp  ./a.out
0x1234
key ^ random = 6b8b4567
key = 0
random = 6b8b4567
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp  ./a.out
0x1234
key ^ random = 6b8b4567
key = 0
random = 6b8b4567
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp  ./a.out
1234
key ^ random = 6b8b41b5
key = 4d2
random = 6b8b4567
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp  ./a.out
1234
key ^ random = 6b8b41b5
key = 4d2
random = 6b8b4567
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp  ./a.out
2345
key ^ random = 6b8b4c4e
key = 929
random = 6b8b4567
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr/tmp
```

[random.c]

```
random@ubuntu:~$ ls
flag  random  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:~$
```

rand()의 return 값을 key라는 지역 변수에 넣는다. == 스택에 남아있겠구나~

[random.c]

```
random@ubuntu:~$ ls
flag  random  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:~$
```

Xor연산을 한 결과가 0xdeadbeef

val1 ^ val2 = val3
&
Val1 ^ val3 = val2

자자. 정답은 집가서 생각해보시고....

*간접적인 풀이만 하겠습니다.

Q&A

http://hackstoryadmin.tistory.com/category/System%20hacking%20training/pwnable.kr

Thank you

```
passcode@ubuntu:~$ file passcode
passcode: setgid ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), dynamically link
ed, interpreter /lib/ld-linux.so.2, for GNU/Linux 2.6.24, BuildID[sha1]=d2b7bd64f70e46b1b0e
b7036b35b24a651c3666b, not stripped
passcode@ubuntu:~$ checksec passcode
[*] '/home/passcode/passcode'
    Arch:     i386-32-little
    RELRO:    Partial RELRO
    Stack:    Canary found
    NX:       NX enabled
    PIE:      No PIE
passcode@ubuntu:~$ exit
logout
Connection to pwnable.kr closed.
 fkillrra@ubuntu  /mnt/hgfs/vm_shared/pwnable.kr  ssh passcode@pwnable.kr -p2222
passcode@pwnable.kr's password:
 ____  __    __  ____    ____  ____   _        ___      __  _  ____
|    \|  |__|  ||    \  /    ||    \ | |      /  _]    |  |/ ]|    \
|  o  )  |  |  ||  _  ||  o  ||  o  )| |     /  [_     |  ' / |  D  )
|   _/|  |  |  ||  |  ||     ||     || |___ |    _]    |    \ |    /
|  |  |  `  '  ||  |  ||  _  ||  O  ||     ||   [_  __ |     \|    \
|  |   \      / |  |  ||  |  ||     ||     ||     ||  ||  .  ||  .  \
|__|    \_/\_/  |__|__||__|__||_____||_____||_____||__||__|\_||__|\_|

- Site admin : daehee87.kr@gmail.com
- IRC : irc.netgarage.org:6667 / #pwnable.kr
- Simply type "irssi" command to join IRC now
- files under /tmp can be erased anytime. make your directory under /tmp
- to use peda, issue `source /usr/share/peda/peda.py` in gdb terminal
Last login: Wed Jul  4 00:11:53 2018 from 176.12.222.219
passcode@ubuntu:~$
```