








MC

32기 김효주






난이도 하 문제를 찾다가 포너블 문제인 MC 문제를 도전해보기로 했다


포너블 문제는 처음이라 어떻게 풀어야 할지 모르겠지만 일단 다운받아서 파일을 열어보자
근데 파일 형식도 안 쓰여 있어서 어떻게 해야 할지 모르겠다

	.gdb_history	2024-07-18 오후 3:15	GDB_HISTORY 파일	1KB
	Dockerfile	2024-07-18 오후 3:03	파일	1KB
	flag	2024-07-18 오후 3:14	파일	1KB
	ld-linux-x86-64.so.2	2024-07-18 오후 3:03	2 파일	236KB
	libc.so.6	2024-07-18 오후 3:03	6 파일	2,165KB
	mc_thread	2024-07-18 오후 3:03	파일	17KB
	mc_thread.c	2024-07-18 오후 3:03	C Source	1KB

파일 중에 ld-linux-x86-64.so가 있으니 이따 리눅스로 열어 봐야겠다

그리고 FLAG 파일이 있길래 메모장으로 열어 확인해 보았다

	.gdb_history	2024-07-18 오후 3:15	GDB_HISTORY 파일	1KB
	Dockerfile	2024-07-18 오후 3:03	파일	1KB
	flag	2024-07-18 오후 3:14	파일	1KB
	ld-linux-x86-64.so.2	2024-07-18 오후 3:03	2 파일	236KB
	libc.so.6	2024-07-18 오후 3:03	6 파일	2,165KB

	◀	.txt	flag	ld-linux-x86-	mc_thread	.gdb_history	flag	×	▶	+
파일	편집	보기								
3S{**flag**}										

이게 플래그일리는 없고... 그냥 플래그 형식을 알려준 것 같다

나머지 파일은 메모장으로 열면 오류가 뜬다

그리고 마지막 파일은 C 파일이길래 VS로 열어서 확인해 보았다

```
1  // Name: mc_thread.c
2  // Compile: gcc -o mc_thread mc_thread.c -pthread -no-pie
3  #include <pthread.h>
4  #include <stdio.h>
5  #include <stdlib.h>
6  #include <unistd.h>
7
8  void giveshell() { execve("/bin/sh", 0, 0); }
9
10 void init() {
11     setvbuf(stdin, 0, 2, 0);
12     setvbuf(stdout, 0, 2, 0);
13 }
14
15 void read_bytes(char *buf, int size) {
16     int i;
17     for (i = 0; i < size; i++)
18         if (read(0, buf + i*8, 8) < 8)
19             return;
20 }
21
22 void thread_routine() {
23     char buf[256];
24     int size = 0;
25     printf("Size: ");
26     scanf("%d", &size);
27     printf("Data: ");
28     read_bytes(buf, size);
29 }
30
31 int main() {
32     pthread_t thread_t;
33
34     init();
35
36     if (pthread_create(&thread_t, NULL, (void *)thread_routine, NULL) < 0) {
37         perror("thread create error:");
38         exit(0);
39     }
40     pthread_join(thread_t, 0);
41 }
```

일단 함수 단위로 분석해보자

- Giveshell() : execve("/bin/sh", 0, 0)를 호출하여 셸을 실행한다. 이 함수를 호출하면 셸을 얻을 수 있다. 그럼 이것 이용해 플래그를 얻을 수 있다는 건가?
- init() : setvbuf를 사용해 버퍼링을 비활성화하거나 라인 버퍼링으로 설정한다고 한다.
- read_bytes() : 사용자 입력을 buf라는 배열에 읽는 함수이다. 한 번에 8바이트씩 읽는다. 이때 사용자가 buf의 크기보다 더 크게 입력할 경우 오버플로우가 발생할 수 있다.
- thread_routine() : scanf로size를 받은 다음, read_bytes 함수를 사용해 데이터를 buf에 채운다. 이때 애도 사용자가 size를 더 크게 입력할 경우 오버플로우가 발생할 수 있다.
- main() : pthread_create를 통해 thread_routine이라는 스레드를 생성한다. pthread_join을 사용하여 메인 스레드가 생성된 스레드가 종료될 때까지 기다린다.

버퍼 오버플로우를 이용하고 giveshell()로 셸을 획득하면 플래그가 나오는 건가? 근데 그걸 어떻게 하는 건지 모르기 때문에 이렇게 분석한 것까지 만족하고... 다음에 포너블을 좀 더 배우면 더 시도해봐야겠다