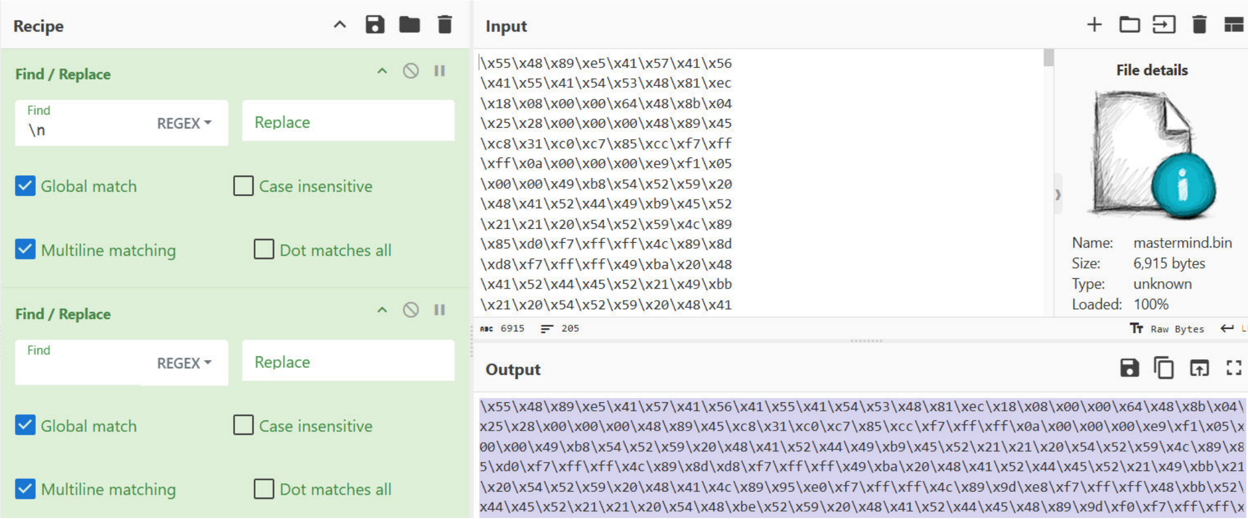
This is one of the more confusing challs when it comes to rev.

Since this is reverse engineering, we quickly come to the conclusion the bytes are probably shellcode, which we need to try to execute.

First, I generated a string containing all the bytes:



Recipe

Find / Replace

Find: \n REGEX
Replace

☑ Global match ☐ Case insensitive

☑ Multiline matching ☐ Dot matches all

Find / Replace

Find: REGEX
Replace

☑ Global match ☐ Case insensitive

☑ Multiline matching ☐ Dot matches all

Input

```
\x55\x48\x89\xe5\x41\x57\x41\x56
\x41\x55\x41\x54\x53\x48\x81\xec
\x18\x08\x00\x00\x64\x48\x8b\x04
\x25\x28\x00\x00\x00\x48\x89\x45
\xc8\x31\xc0\xc7\x85\xcc\xf7\xff
\xff\x0a\x00\x00\x00\xe9\xf1\x05
\x00\x00\x49\xb8\x54\x52\x59\x20
\x48\x41\x52\x44\x49\xb9\x45\x52
\x21\x21\x20\x54\x52\x59\x4c\x89
\x85\xd0\xf7\xff\xff\x4c\x89\x8d
\xd8\xf7\xff\xff\x49\xba\x20\x48
\x41\x52\x44\x45\x52\x21\x49\xbb
\x21\x20\x54\x52\x59\x20\x48\x41
```

File details

Name: mastermind.bin
Size: 6,915 bytes
Type: unknown
Loaded: 100%

Output

```
\x55\x48\x89\xe5\x41\x57\x41\x56\x41\x55\x41\x54\x53\x48\x81\xec\x18\x08\x00\x00\x64\x48\x8b\x04\
x25\x28\x00\x00\x00\x48\x89\x45\xc8\x31\xc0\xc7\x85\xcc\xf7\xff\xff\x0a\x00\x00\x00\xe9\xf1\x05\x
00\x00\x49\xb8\x54\x52\x59\x20\x48\x41\x52\x44\x49\xb9\x45\x52\x21\x21\x20\x54\x52\x59\x4c\x89\x8
5\xd0\xf7\xff\xff\x4c\x89\x8d\xd8\xf7\xff\xff\x49\xba\x20\x48\x41\x52\x44\x45\x52\x21\x49\xbb\x21
\x20\x54\x52\x59\x20\x48\x41\x4c\x89\x95\xe0\xf7\xff\xff\x4c\x89\x9d\xe8\xf7\xff\xff\x48\xbb\x52\
x44\x45\x52\x21\x21\x20\x54\x48\xbe\x52\x59\x20\x48\x41\x52\x44\x45\x48\x89\x9d\xf0\xf7\xff\xff\x
```

Next, I found a short C code that executes the shellcode

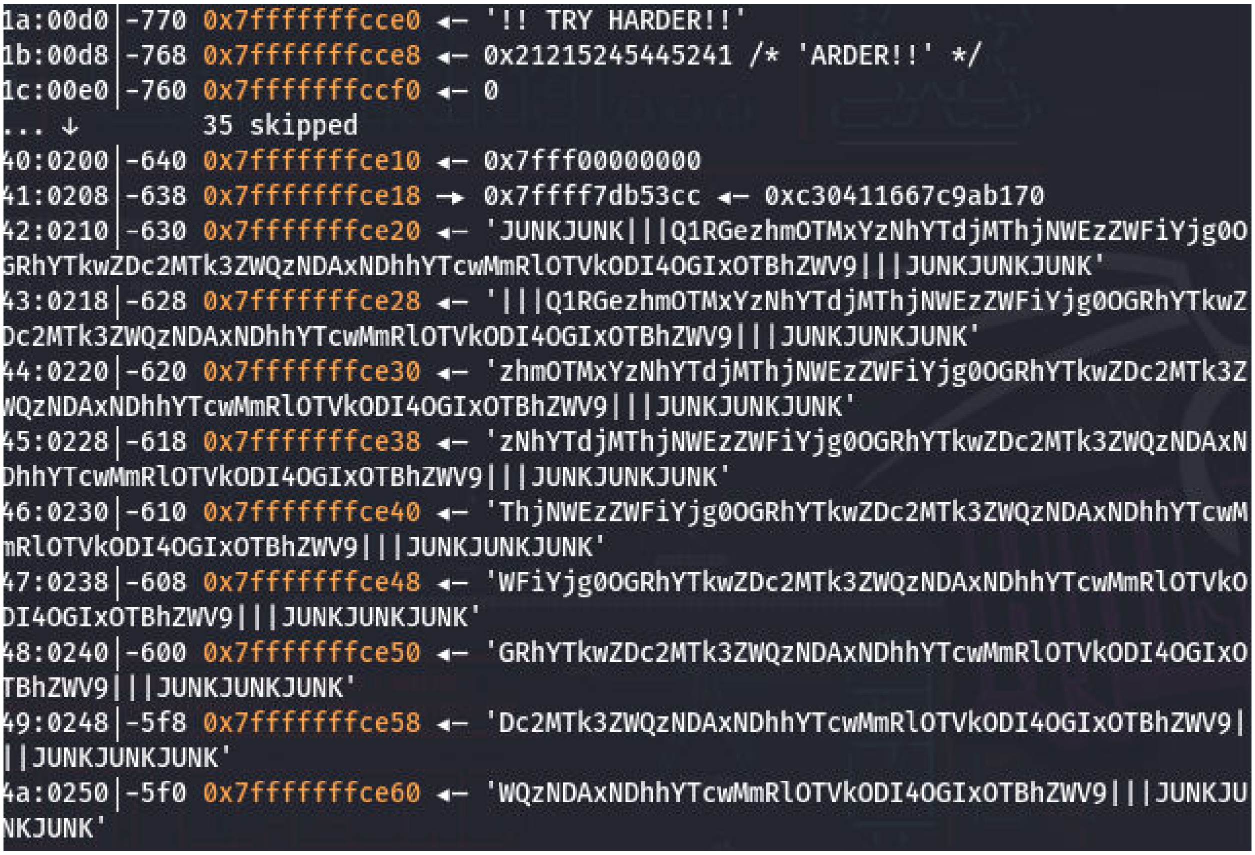
```c
#include <stdio.h>
#include <string.h>
#include <unistd.h>
#include <sys/mman.h>

int main() {
    unsigned char shellcode[] = "copy_bytes_here";
    void *exec_mem = mmap(NULL, sizeof(shellcode), PROT_READ | PROT_WRITE |
PROT_EXEC, MAP_PRIVATE | MAP_ANONYMOUS, -1, 0);
    memcpy(exec_mem, shellcode, sizeof(shellcode));
    ((void (*)())exec_mem)();
    munmap(exec_mem, sizeof(shellcode));
    return 0;
}
```

Then I compiled it on my linux machine and ran it. But nothing was happening.

As a result, I ran it in gdb, where again nothing was happening for a while

I quit the program with CTRL + C and saw a lot of TRY HARDER stuff (my pwndbg shows me stuff upon stopping the program)

Looking through the stack I found some strange stuff:



```
1a:00d0│-770 0x7fffffffcce0 ◂— '!! TRY HARDER!!'
1b:00d8│-768 0x7fffffffcce8 ◂— 0x21215245445241 /* 'ARDER!!' */
1c:00e0│-760 0x7fffffffccf0 ◂— 0
... ↓        35 skipped
40:0200│-640 0x7fffffffce10 ◂— 0x7fff00000000
41:0208│-638 0x7fffffffce18 —▸ 0x7ffff7db53cc ◂— 0xc30411667c9ab170
42:0210│-630 0x7fffffffce20 ◂— 'JUNKJUNK|||Q1RGezhmOTMxYzNhYTdjMThjNWEzZWFiYjg0O
GRhYTkwZDc2MTk3ZWQzNDAxNDhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
43:0218│-628 0x7fffffffce28 ◂— '|||Q1RGezhmOTMxYzNhYTdjMThjNWEzZWFiYjg0OGRhYTkwZ
Dc2MTk3ZWQzNDAxNDhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
44:0220│-620 0x7fffffffce30 ◂— 'zhmOTMxYzNhYTdjMThjNWEzZWFiYjg0OGRhYTkwZDc2MTk3Z
WQzNDAxNDhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
45:0228│-618 0x7fffffffce38 ◂— 'zNhYTdjMThjNWEzZWFiYjg0OGRhYTkwZDc2MTk3ZWQzNDAxN
DhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
46:0230│-610 0x7fffffffce40 ◂— 'ThjNWEzZWFiYjg0OGRhYTkwZDc2MTk3ZWQzNDAxNDhhYTcwM
mRlOTVkODI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
47:0238│-608 0x7fffffffce48 ◂— 'WFiYjg0OGRhYTkwZDc2MTk3ZWQzNDAxNDhhYTcwMmRlOTVkO
DI4OGIxOTBhZWV9|||JUNKJUNKJUNK'
48:0240│-600 0x7fffffffce50 ◂— 'GRhYTkwZDc2MTk3ZWQzNDAxNDhhYTcwMmRlOTVkODI4OGIxO
TBhZWV9|||JUNKJUNKJUNK'
49:0248│-5f8 0x7fffffffce58 ◂— 'Dc2MTk3ZWQzNDAxNDhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|
||JUNKJUNKJUNK'
4a:0250│-5f0 0x7fffffffce60 ◂— 'WQzNDAxNDhhYTcwMmRlOTVkODI4OGIxOTBhZWV9|||JUNKJU
NKJUNK'
```

Turns out, the first JUNKJUNK string in the picture is the flag in base64. So…. We got the flag I guess.

Made with love by: AndreiCat