Baby C my mind

Overview

This challenge implies founding the key and iv for the aes cbc encryption used for encrypting the so file that a has the flag in it.

1 Dumping the encrypted so file

To dump the encrypted so file you need pwndbg to set the breakpoint at main function and from there to run the program untill you see that something is load in memory with dl.Pwndbg will show you the pid and the filedesciptor of the encrypted so file, with which you can dump it with cp /proc/15989/fd/3 /Desktop/cmymind.bin

2 Finding the key and iv

After that you will see in the decompiled code of the original elf that the encrypted blob was encrypted with aes.In the decompiled code you will see this array,the first 16 integers represent the key and the last 16 the iv.With them you can decrypt the encrypted.bin and find the flag.

```
3       encryptiv((__int64)v20);
4       v20[16] = 1;
5       v20[17] = 2;
6       v20[18] = 4;
7       v20[19] = 2;
8       v20[20] = 3;
9       v20[21] = 4;
0       v20[22] = 4;
1       v20[23] = 2;
2       v20[24] = 3;
3       v20[25] = 2;
4       v20[26] = 4;
5       v20[27] = 3;
6       v20[28] = 2;
7       v20[29] = 3;
8       v20[30] = 4;
9       v20[31] = 3;
0       v20[32] = 8;
1       v20[33] = 9;
2       v20[34] = 2;
3       v20[35] = 3;
4       v20[36] = 4;
5       v20[37] = 7;
6       v20[38] = 9;
7       v20[39] = 4;
8       v20[40] = 4;
9       v20[41] = 7;
0       v20[42] = 7;
1       v20[43] = 8;
2       v20[44] = 8;
3       v20[45] = 7;
4       v20[46] = 4;
5       v20[47] = 9;
6       aes_encrypt();
7       v16 = (void (*)(void))dlsym(handle, "flag");
```

3 UVT{at_a3s_w3_th1nk_wh3n_m3mory_1s_ly1ng}

4 Thanks for reading this write-up!