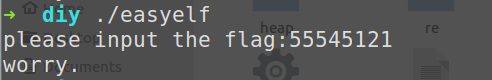
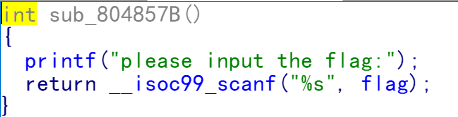
是个elf文件，在linux下运行，程序功能就是判断你输入的flag是否正确



拖进ida,主函数不能直接F5生成伪C代码。于是直接看汇编

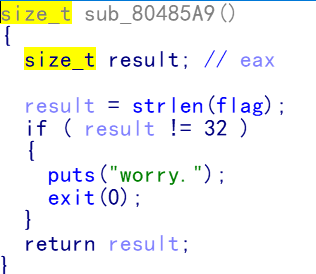
一大堆重复花指令，一行一行慢慢往下看。在花指令中掺杂着几个重要函数。

1.sub\_804857B()



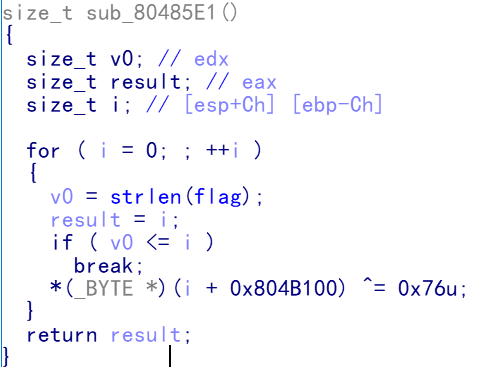
输入函数，将输入保存在flag字符串中

2.sub\_80485A9()



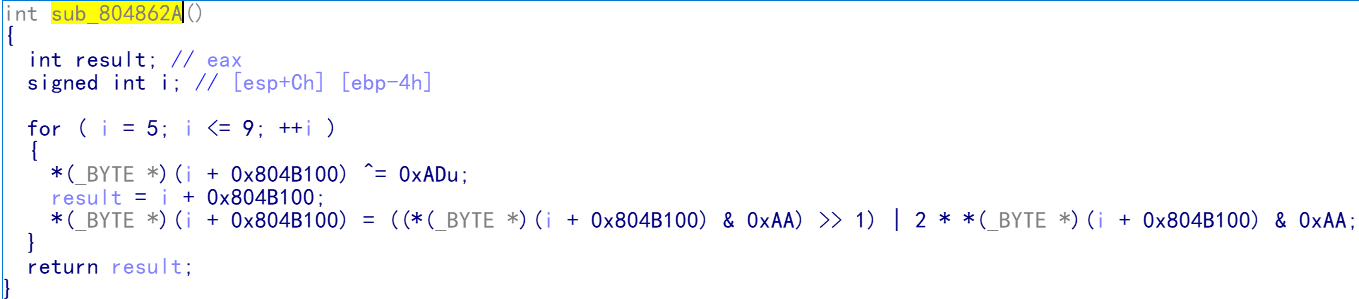
判断flag长度是否为32，由此可知flag长度为32

3. sub\_80485E1()



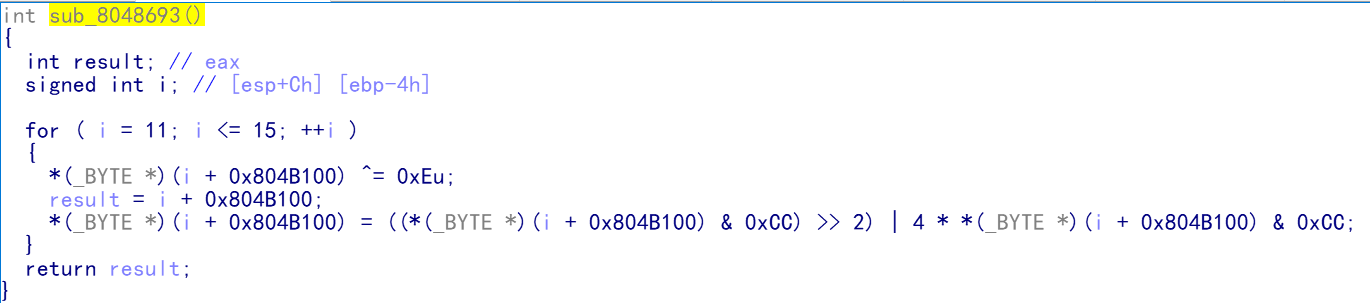
将flag每个字符和0x76异或

4. sub\_804862A()



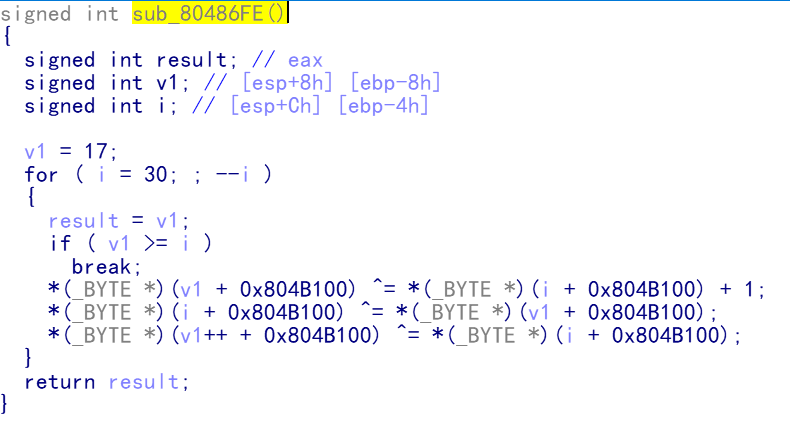
将flag的6-10个字符进行处理

5. sub\_8048693()



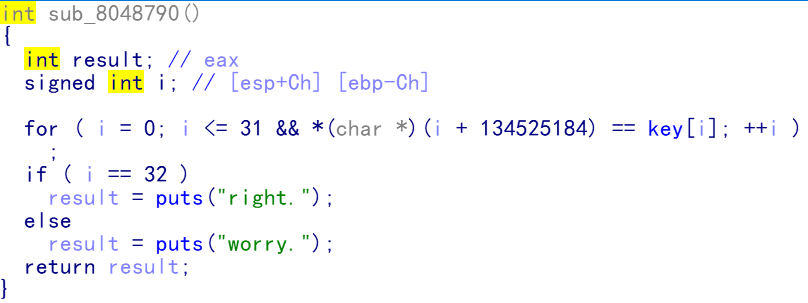
将flag的12-16个字符进行处理

6. sub\_80486FE()



将flag的18-31个字符进行处理

7. sub\_8048790()



将经过一系列处理的flag和key数组进行比较，相同则输出right.

总的流程分析到此，写脚本分段进行爆破，最终得到flag.脚本见附件exp.py