**汇编语言与逆向技术实验报告**

**Lab6 - RE Challenge #2**

**学号：2112492 姓名：刘修铭 专业：信息安全**

**一、实验目的**

1.熟悉静态反汇编工具IDA Freeware；

2.熟悉反汇编代码、伪代码的逆向分析过程；

3.掌握反汇编语言中的数学计算、数据结构、条件判断、分支结构的识别和逆向分析；

4.运用熟悉的编程语言，实现简单的脚本编写。

**二、实验原理**

1.通过IDA Freeware得到ELF文件的反汇编代码，如图1和图2所示。



图 1 reverse的反汇编代码

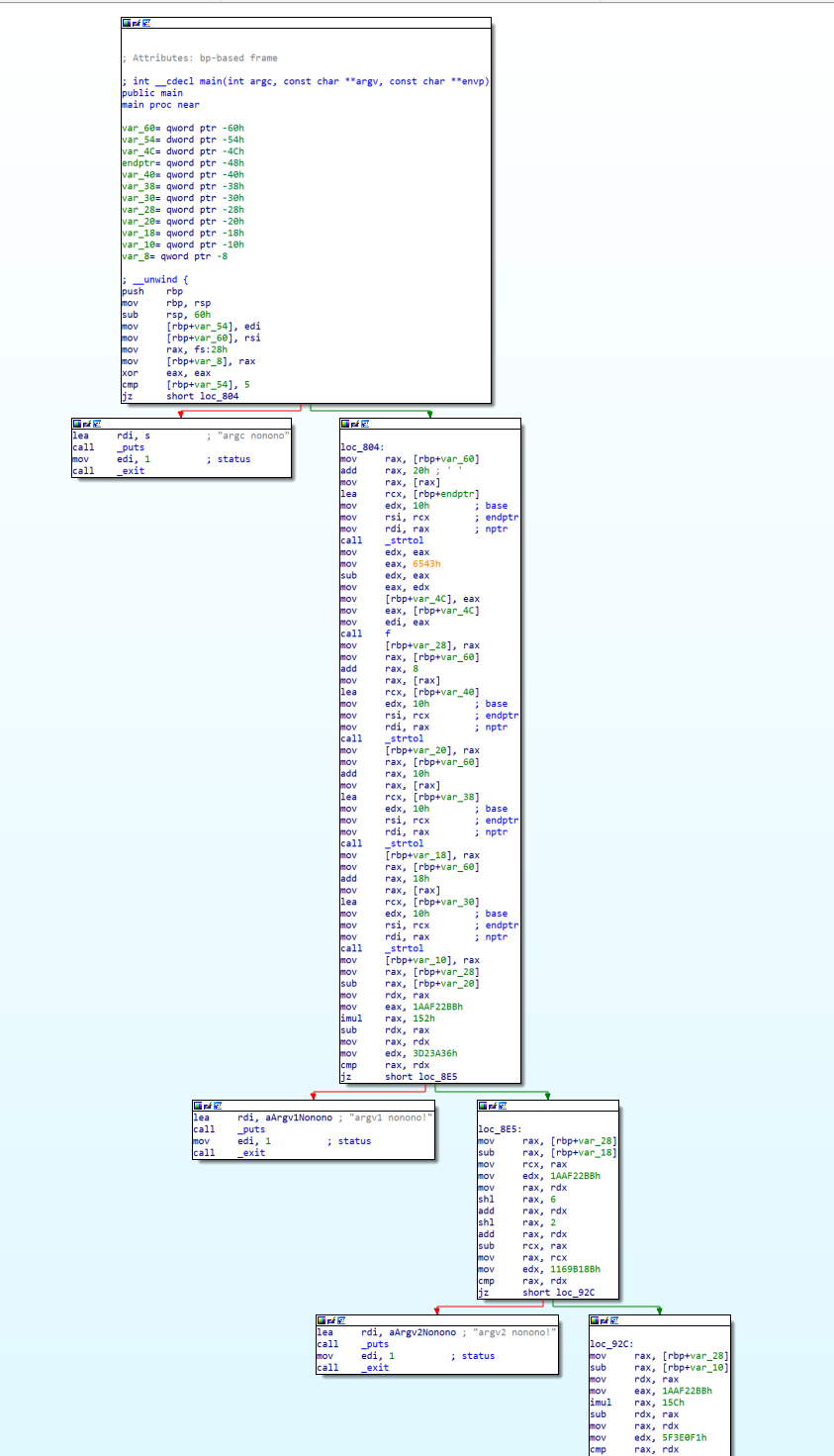


图 2 reverse的反汇编代码的图形化显示

2.使用IDA的反编译功能（F5）得到伪代码，如图3所示。

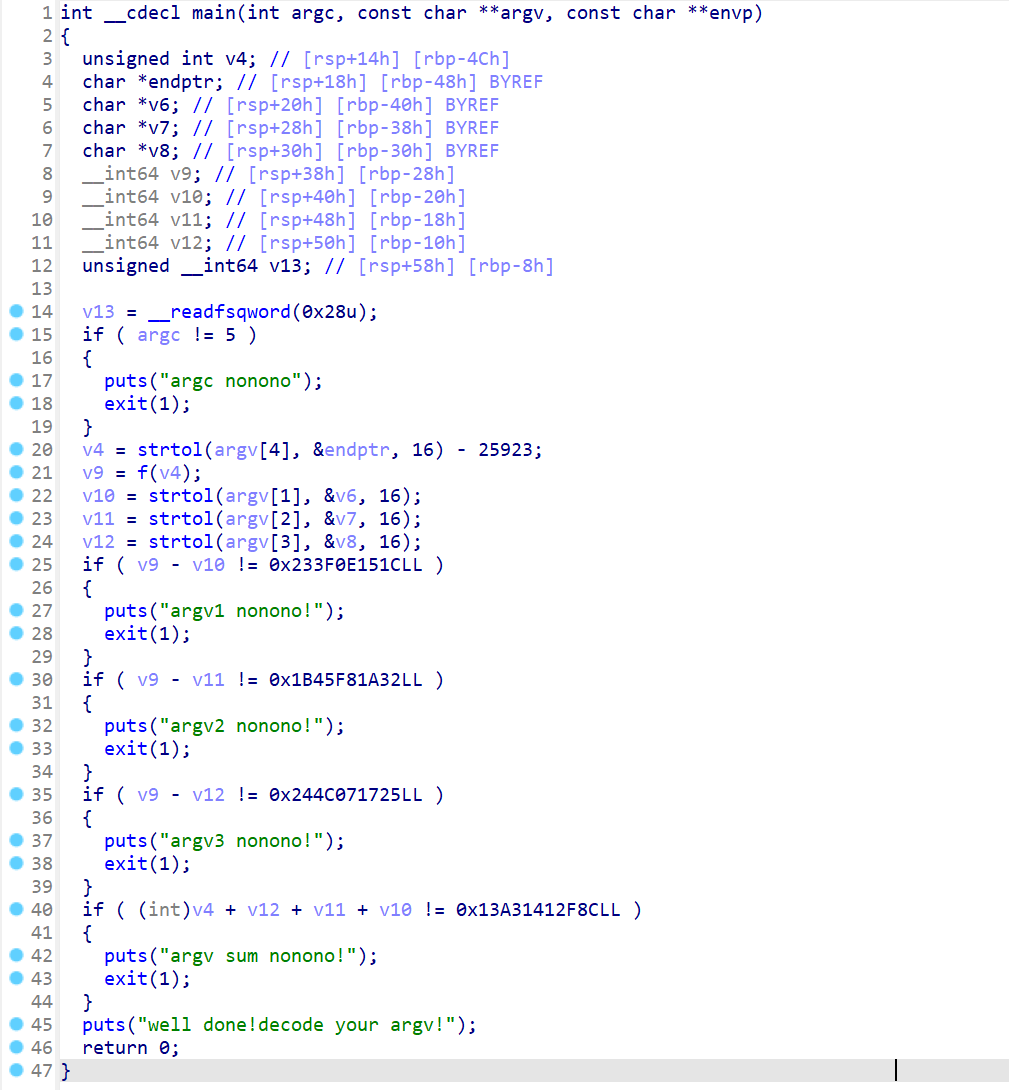


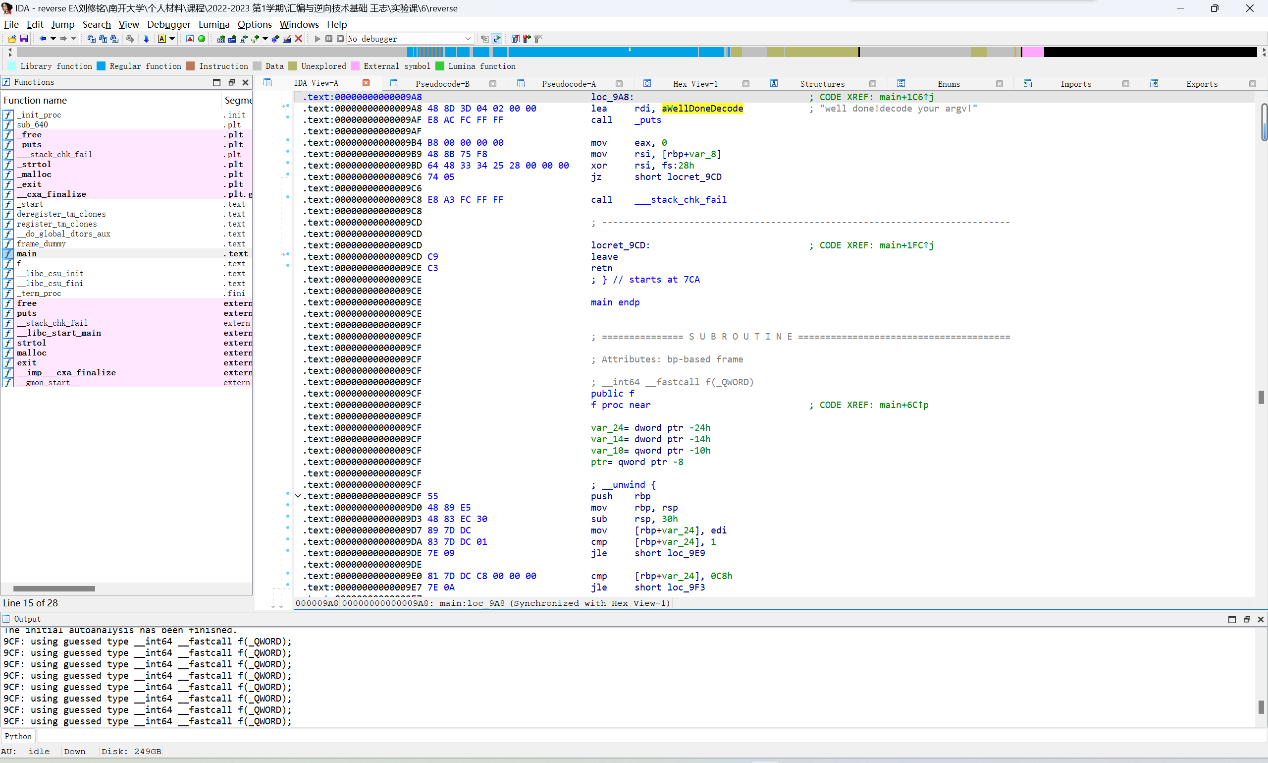
图 3 reverse的反编译伪代码

3.对汇编代码和反编译伪代码的计算过程、条件判断、分支结构等信息进行分析，逆向推出待解方程组。

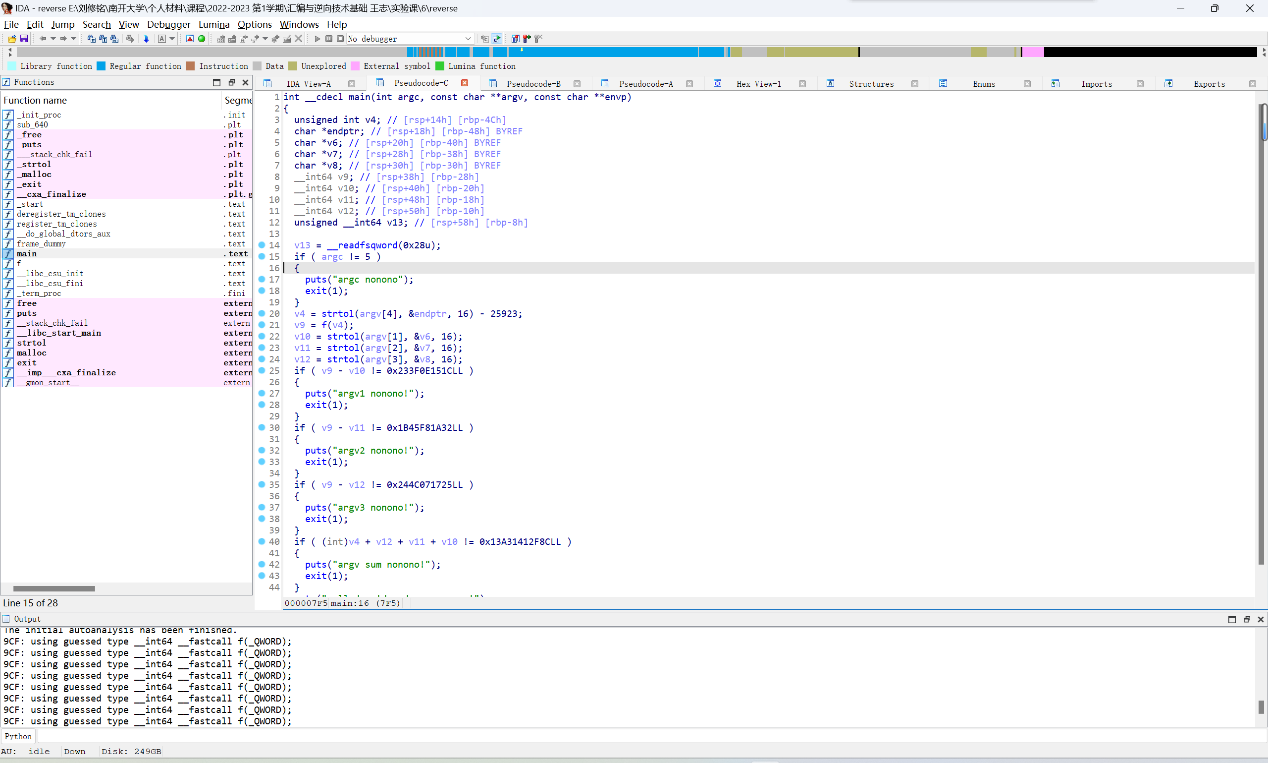
4.编写脚本实现暴力破解，解出方程组，得到参数“V9”、“V10”、“V11”、“V12”的正确取值，完成逆向分析挑战！

**三、获取反汇编代码**

通过IDA Freeware可以得到二进制代码的反汇编代码及反汇编伪代码：



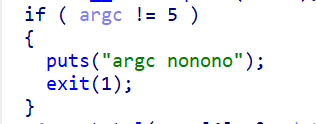
反汇编代码



反汇编伪代码

**四、逆向过程分析**

**1.“argc nonono”**



此代码块的功能是：

* 将argc和5比较
* 若 输入的参数个数为4，进入下一步；否则报错"argc nonono"

**2.v4**



此代码块的功能是：

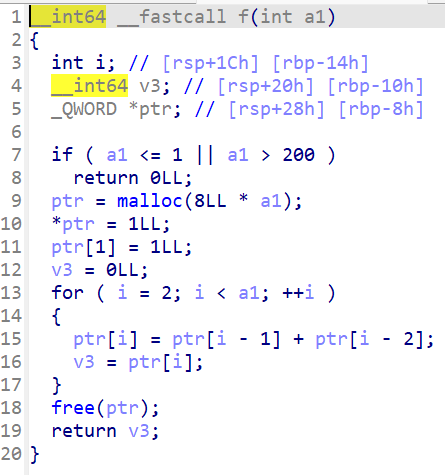
* 将输入的第四个参数（十六进制）减去0x6543后存入v4

**3.v9**



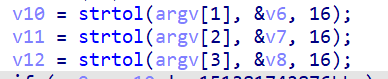
此代码块的功能是：

* 将v4的值作为参数调用函数f，并将函数的返回值赋值给v9



* 查阅函数f的代码后可知，函数f为斐波那契数列的求值函数，其中参数v4表示加和次数。

**4.v10、v11、v12分别表示传入的参数的第一、二、三位（十六进制）**



此代码块的功能是：

* 将v10、v11、v12分别传入参数的第一、二、三位（十六进制）

**5.if语句**



此代码块的功能是：

* 将v10、v11、v12（十六进制）分别与v9进行运算，得到相应的值

综上，得到如下运算关系：

f(a4-0x6543) - a1      = 0x233F0E151C

f(a4-0x6543) - a2      = 0x1B45F81A32

f(a4-0x6543) - a3      = 0x244C071725

a4-0x6543 + a3 + a2 + a1 = 0x13A31412F8C

a4-0x6543 + 3\* fib(a4-0x6543) = sum()

**五、参数求解**

由逆向分析可知，需结合斐波那契数列的求和公式与原始代码中的限定关系联合求解四个参数。

脚本代码如下：

def f(n):

    a=1

    b=1

    i=2

    while i<n:

       a,b = b,a+b

       i+=1

    return b

sum = 0x233F0E151C + 0x1B45F81A32 + 0x244C071725 + 0x13A31412F8C

for v4 in range(58,201):

    v9 = f(v4)

    if v4+3\*v9 == sum:

        print(v4)

        print(bytes.fromhex(hex(v9-0x233F0E151C)[2:])+ bytes.fromhex(hex(v9-0x1B45F81A32)[2:]) +bytes.fromhex(hex(v9-0x244C071725)[2:]) + bytes.fromhex(hex(v4+0x6543)[2:]))

        break

运行可得。

故

由四个if语句可知：

即，四个参数的值分别为：