

**实验报告**

**实 验（三）**

题 目 Binary Bomb

二进制炸弹

专 业 计算机专业

学　　 号 1190200501

班　　 级 1903002

学 生 林燕燕

指 导 教 师 郑贵滨

实 验 地 点 G709

实 验 日 期 2021.04.23

**计算机科学与技术学院**

**目 录**

[第1章 实验基本信息 - 3 -](#_Toc496608242)

[1.1 实验目的 - 3 -](#_Toc496608243)

[1.2 实验环境与工具 - 3 -](#_Toc496608244)

[1.2.1 硬件环境 - 3 -](#_Toc496608245)

[1.2.2 软件环境 - 3 -](#_Toc496608246)

[1.2.3 开发工具 - 3 -](#_Toc496608247)

[1.3 实验预习 - 3 -](#_Toc496608248)

[第2章 实验环境建立 - 4 -](#_Toc496608249)

[2.1 Ubuntu下CodeBlocks反汇编（10分） - 4 -](#_Toc496608250)

[2.2 Ubuntu下EDB运行环境建立（10分） - 4 -](#_Toc496608251)

[第3章 各阶段炸弹破解与分析 - 5 -](#_Toc496608252)

[3.1 阶段1的破解与分析 - 5 -](#_Toc496608253)

[3.2 阶段2的破解与分析 - 6 -](#_Toc496608254)

[3.3 阶段3的破解与分析 - 7 -](#_Toc496608255)

[3.4 阶段4的破解与分析 - 8 -](#_Toc496608256)

[3.5 阶段5的破解与分析 - 9 -](#_Toc496608257)

[3.6 阶段6的破解与分析 - 9 -](#_Toc496608258)

[3.7 阶段7的破解与分析(隐藏阶段) - 9 -](#_Toc496608259)

[第4章 总结 - 10 -](#_Toc496608260)

[4.1 请总结本次实验的收获 - 10 -](#_Toc496608261)

[4.2 请给出对本次实验内容的建议 - 10 -](#_Toc496608262)

[参考文献 - 11 -](#_Toc496608263)

# 第1章 实验基本信息

## 1.1 实验目的

熟练掌握计算机系统的ISA指令系统与寻址方式

熟练掌握Linux下调试器的反汇编调试跟踪分析机器语言的方法

增强对程序机器级表示、汇编语言、调试器和逆向工程等的理解

## 1.2 实验环境与工具

### 1.2.1 硬件环境

X64 CPU；1.6GHz；8G RAM；256G SSD Disk；1T HDD Disk

### 1.2.2 软件环境

Windows10 64位；Vmware 14pro；Ubuntu 20.04.2 LTS 64位

### 1.2.3 开发工具

Visual Studio Code 64位；vim/gpedit+gcc

## 1.3 实验预习

* 请写出C语言下包含字符串比较、循环、分支（含switch）、函数调用、递归、指针、结构、链表等的例子程序sample.c。
* 生成执行程序sample.out。
* 用gcc –S或CodeBlocks或GDB或OBJDUMP等，反汇编，比较。
* 列出每一部分的C语言对应的汇编语言。
* 修改编译选项-O (缺省2)、O0、O1、O2、O3，-m32/m64。再次查看生成的汇编语言与原来的区别。
* 注意O1之后无栈帧，EBP做别的用途。-fno-omit-frame-pointer加上栈指针。
* GDB命令详解 –tui模式 ^XA切换 layout改变等等
* 有目的地学习: 看VS的功能GDB命令用什么？

# 第2章 实验环境建立

## 2.1 Ubuntu下CodeBlocks反汇编（10分）

CodeBlocks运行hellolinux.c。反汇编查看printf函数的实现。

要求：C、ASM、内存(显示hello等内容)、堆栈（call printf前）、寄存器同时在一个窗口。

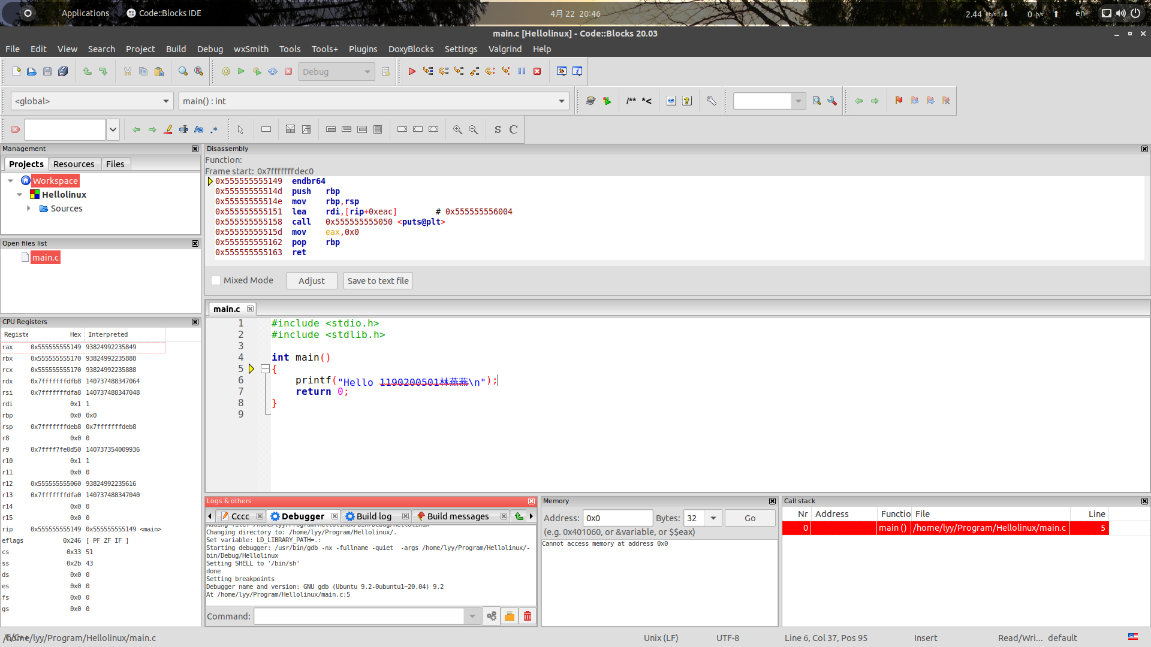


图2-1 Ubuntu下CodeBlocks反汇编截图

## 2.2 Ubuntu下EDB运行环境建立（10分）

用EDB调试hellolinux.c的执行文件，截图，要求同2.1

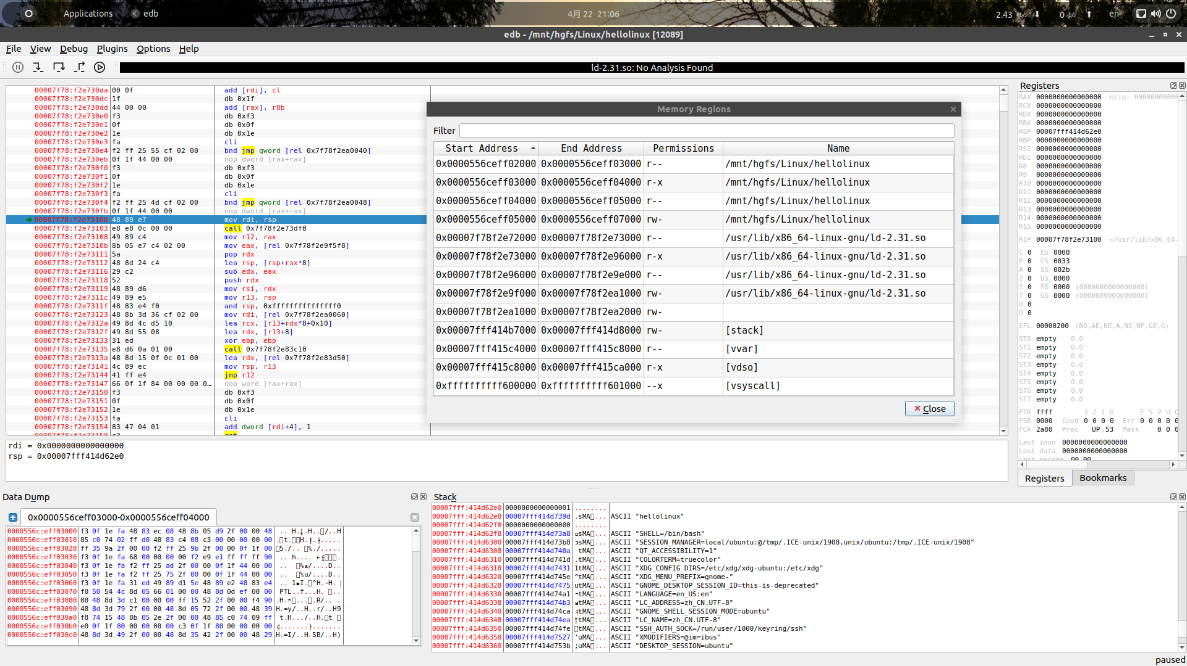


图2-2 Ubuntu下EDB截图

# 第3章 各阶段炸弹破解与分析

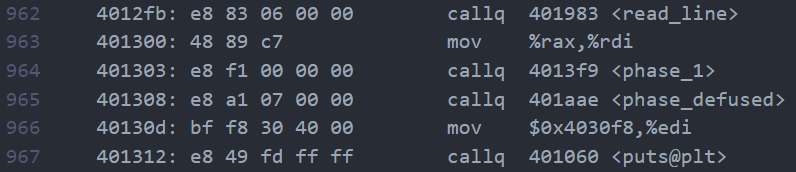
每阶段15分（密码10分，分析5分），总分不超过80分

## 3.1 阶段1的破解与分析

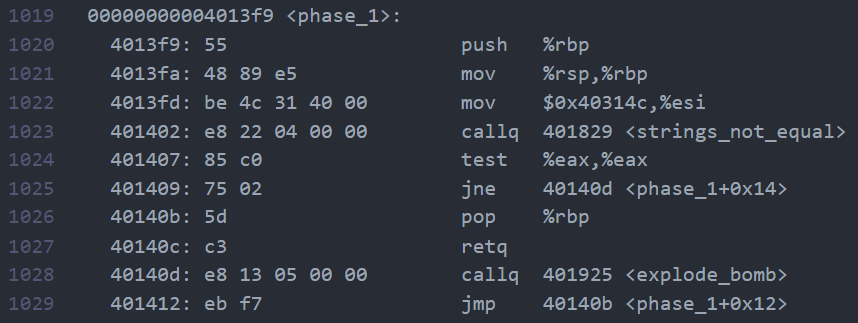
密码如下：Public speaking is very easy.

破解过程：

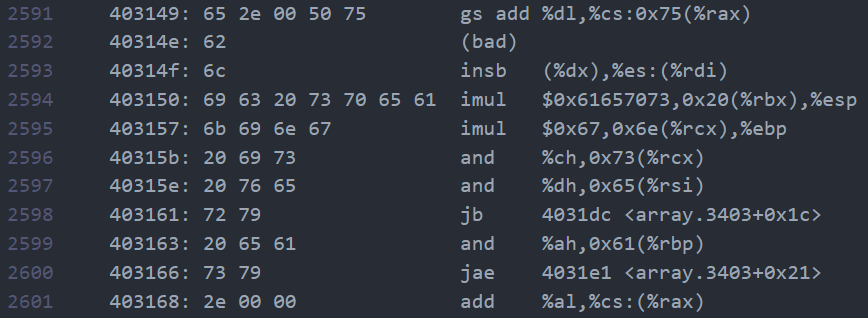
对bomb进行反汇编得到汇编代码，查看代码在main函数中找到phase\_1函数位置：



继续查找phase\_1的位置：



从键盘输入或文件读取的字符串存入rdi，esi存放用于比对的字符串，地址在0x40314c： 

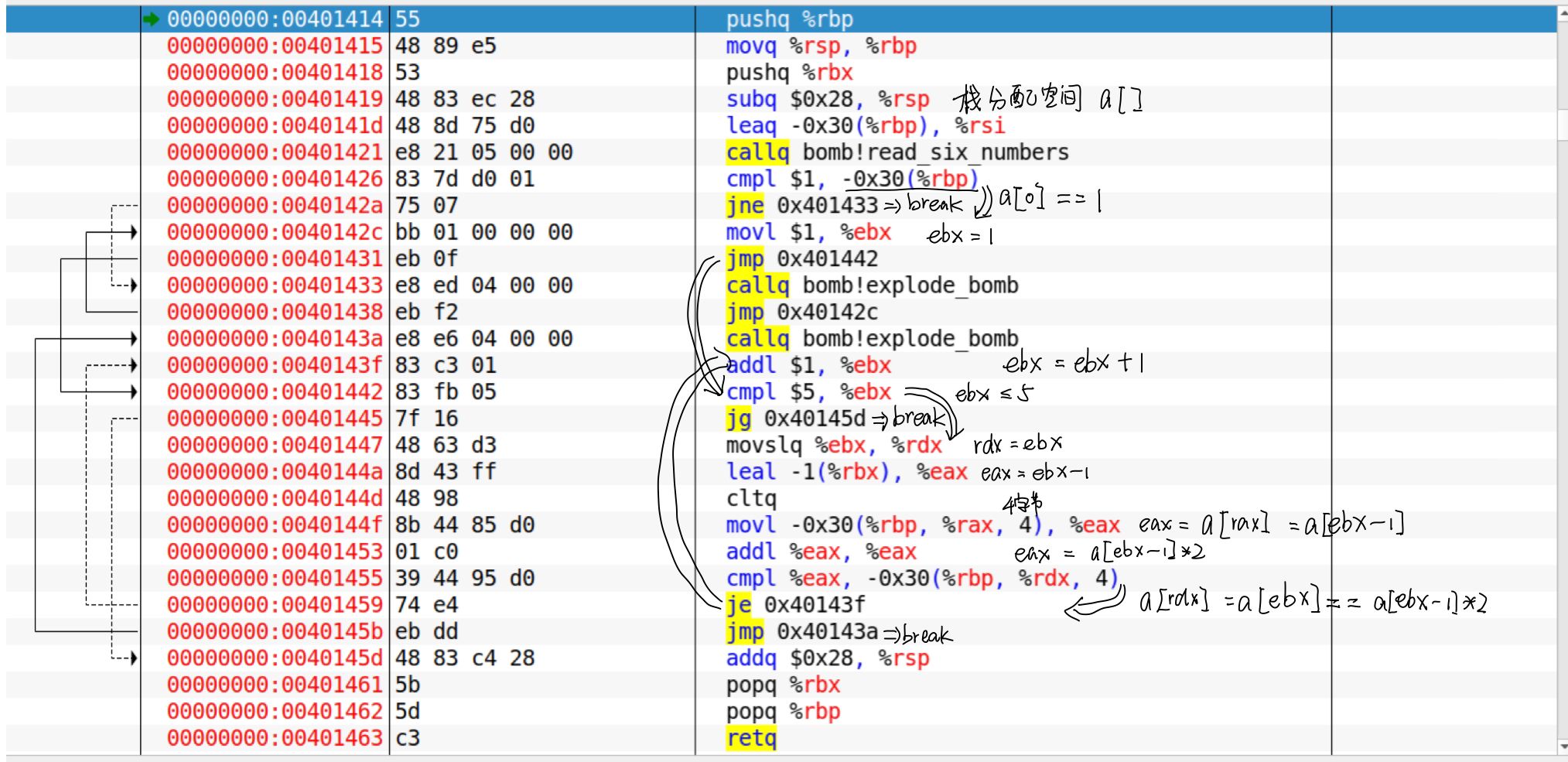


由16进制机器码可得esi存放字符串十六进制数为50 75 62 6c 69 63 20 73 70 65 61 6b 69 6e 67 20 69 73 20 76 65 72 79 20 65 61 73 79 2e 00 00，转换为ASCII码得到 Public speaking is very easy. ，即为密码。

## 3.2 阶段2的破解与分析

密码如下：1 2 4 8 16 32

破解过程：

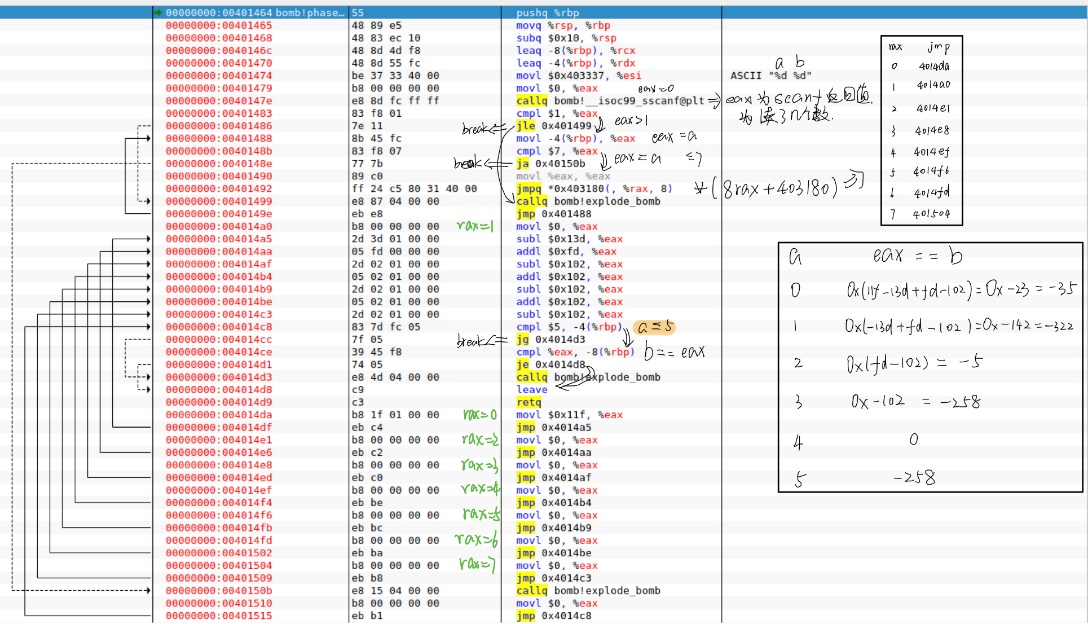


phase2中要求输入6个数，输入采用循环，且存在 表示后一个数为前一个数的两倍，又有 表示第一个数为 1，则密码为：1 2 4 8 16 32

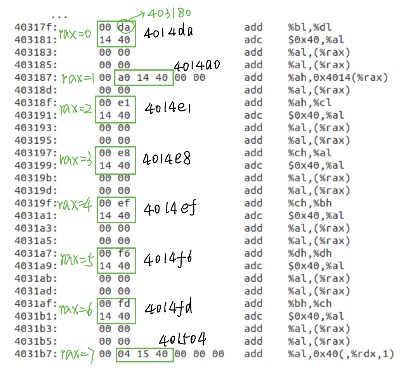
## 3.3 阶段3的破解与分析

密码如下：0 -35 或 1 -322 或 2 -5 或 3 -258 或 4 0 或 5 -258 (六组密码)

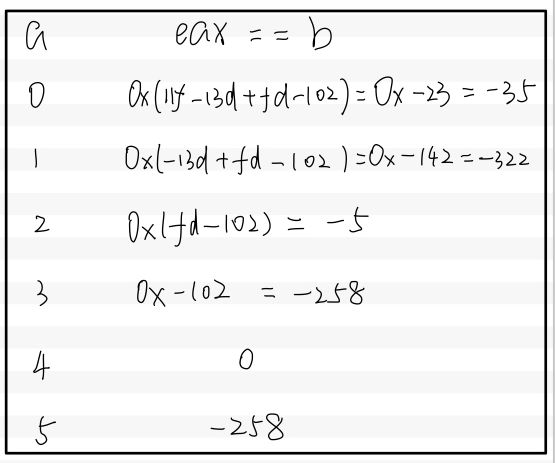
破解过程：



phase3中输入两个数字a b，a要求小于等于7即0~7，有 计算数据地址，找到代码段，如下图，rax = a



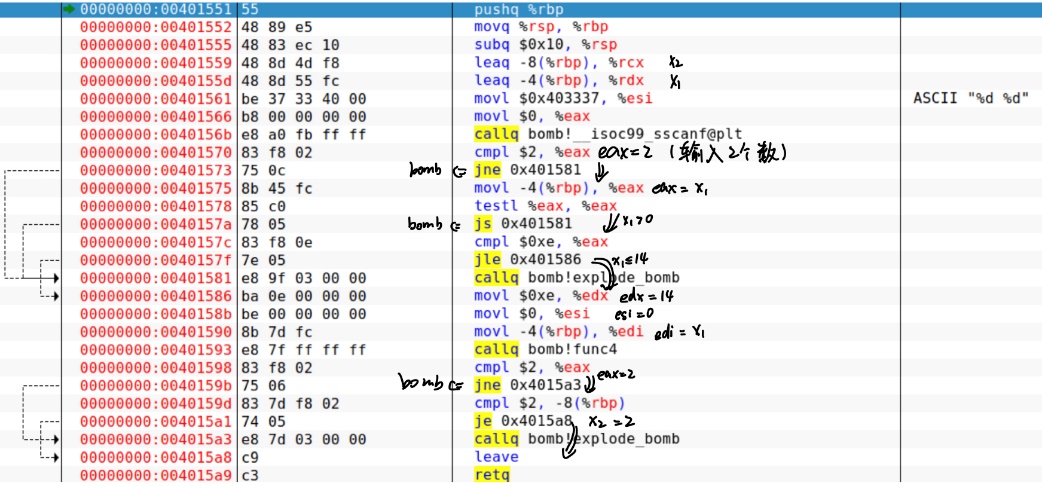
得到跳转地址，又有  ，则 a <= 5 ，得到密码有：

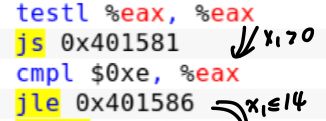
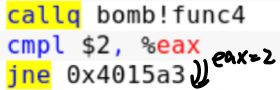
共六组密码。

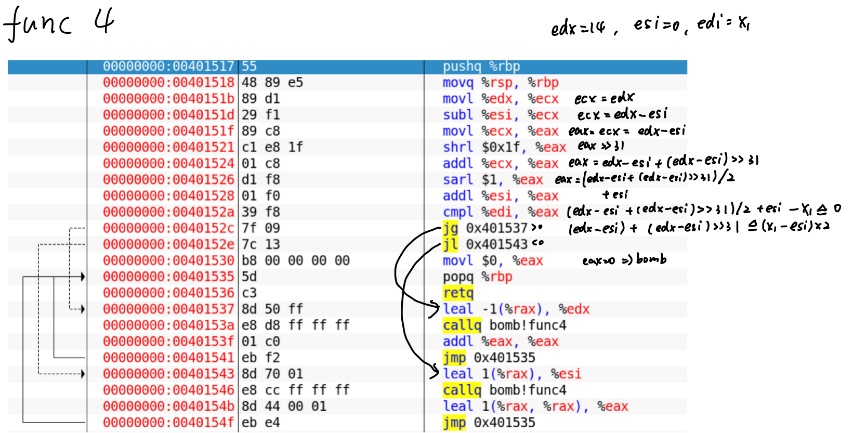
## 3.4 阶段4的破解与分析

密码如下：4 2 或 5 2 (两组密码)

破解过程：



phase4中输入两个数x1 x2，x1要大于0且小于等于14 ，接着对edx esi edi 赋值，传func4，而func4的返回值要是2。



进入func4，计算得到，将eax与edi即x1进行比较，按情况赋值进行递归，测试代码如下

**#include** <stdio.h>

int x1;

int eax;

int **func4**(int edx,int esi){

    eax = ((edx - esi + (((unsigned)(edx - esi))>>31))>>1) + esi;

    int text = eax - x1;

**if**(text **>** 0)    {

        edx = eax - 1;

**func4**(edx,esi);

        eax = eax \* 2;

    }

**else** **if**(text **<** 0){

        esi = eax + 1;

**func4**(edx,esi);

        eax = 1 + eax \* 2;

    }

**else**{

        eax = 0;

    }

**return** 0;

}

int **main**(){

**for**(x1 = 1;x1 **<=** 14;x1++)    {

**func4**(14,0);

**if**(eax **==** 2){

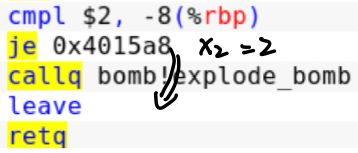
**printf**("x1 = %d\n",x1);

        }

    }

**return** 0;

}

返回eax需为2，计算结果为，即x1=4或5，又，即x2 = 2，则密码为4 2, 5 2 两组。

## 3.5 阶段5的破解与分析

密码如下：

破解过程：

## 3.6 阶段6的破解与分析

密码如下：

破解过程：

## 3.7 阶段7的破解与分析(隐藏阶段)

密码如下：

破解过程：

# 第4章 总结

## 4.1 请总结本次实验的收获

本次实验使我熟悉了计算机系统的ISA指令系统与寻址方式，加深了对反汇编程序的理解。在拆弹的过程中更加深刻地学习了各种反汇编代码，拆弹过程也很有趣。

## 4.2 请给出对本次实验内容的建议

希望可以更有趣些。

注：本章为酌情加分项。

# 参考文献

[1] 林来兴. 空间控制技术[M]. 北京：中国宇航出版社，1992：25-42.

[2] 辛希孟. 信息技术与信息服务国际研讨会论文集：A集[C]. 北京：中国科学出版社，1999.

[3] 赵耀东. 新时代的工业工程师[M/OL]. 台北：天下文化出版社，1998 [1998-09-26]. http://www.ie.nthu.edu.tw/info/ie.newie.htm（Big5）.

[4] 谌颖. 空间交会控制理论与方法研究[D]. 哈尔滨：哈尔滨工业大学，1992：8-13.

[5] KANAMORI H. Shaking Without Quaking[J]. Science，1998，279（5359）：2063-2064.

[6] CHRISTINE M. Plant Physiology: Plant Biology in the Genome Era[J/OL]. Science，1998，281：331-332[1998-09-23]. http://www.sciencemag.org/cgi/ collection/anatmorp.