实 验 一 ：x86 汇编基础-二进制炸弹

（完成时间：第七周）

一、 实验目的

⑴ 初步认识 X86 汇编语言；
⑵ 掌握阅读程序反汇编代码的方法，了解程序在机器上运行的实质；
⑶ 熟悉 Linux 环境、掌握调试器 gdb 和反汇编工具 objdump 的使用。

二、 实验内容

使用课程知识拆除一个“Binary Bomb” （，简称炸弹）来增强对程序的机器级表示、汇编语言、调试器和逆向工程等理解。二进制炸弹是一个 Linux 可执行 C 程序，包含 phase_1~phase_6 共 6 个阶段和一个隐藏阶段 secret_phase。你将获得一个唯一且每位同学差异化的炸弹程序。炸弹运行各阶段要求输入一个字符串，若输入符合程序预期，该阶段炸弹被“拆除”，否则“爆炸”。实验目标是你需要拆除尽可能多的炸弹。

每个炸弹阶段考察机器级语言程序不同方面，难度递增。

阶段 1：字符串比较

阶段 2：循环

阶段 3：条件/分支：含 switch 语句

阶段 4：递归调用和栈

阶段 5：指针

阶段 6：链表/指针/结构

隐藏阶段，第 4 阶段的之后附加一特定字符串后才会出现

拆弹技术：为了完成二进制炸弹拆除任务，你需要

1.使用 gdb 调试器和 objdump 反汇编工具；

2.单步跟踪调试每一阶段的机器代码

3.理解汇编语言代码的行为或作用

4.进而设法“推断”出拆除炸弹所需的目标字符串。

5.需要在每一阶段的开始代码前和引爆炸弹的函数前设置断点，便于调试。

三、实验指导

1.炸弹文件包包含文件：

bomb：bomb 的可执行程序。

bomb.c：bomb 程序的 main 函数。

ID：你的学号。

README：用文本编辑器打开即可查看其内容。

bomb：是一个 linux 下可执行程序，需要 0 或 1 个命令行参数（详见 bomb.c 源文件中的 main()函数）。运行时不指定参数，则该程序打印出欢迎信息后，期待你按行输入每一阶段用来拆除炸弹的字符串，并根据你当前输入的字符串决定你是通过相应阶段还是炸弹爆炸导致任务失败。

bomb.c：bomb 主程序，不是全部，看不到炸弹，只能看到程序的主要框架。

2.实验结果及结果文件

可在命令行运行 bomb，然后根据提示，逐阶段输入拆弹字符串。

也可将拆除每一阶段炸弹的字符串按行组织在一个文本文件中，如 solution.txt，然后作为参数传给程序。

结果文件格式：每个拆弹字符串一行，回车结束，最多 7 行（包含最后特殊阶段），除此之外不要包含任何其它字符。范例如下：

```
string1
string2
……
string6
string7
```

实验结果文件使用方法（在命令行中国输入）：./bomb solution.txt

程序会自动读取文本文件中的字符串，并依次检查对应每一阶段的字符串来决定炸弹拆除成败。

四、实验设备

PC 机一台，装有 Linux 操作系统的虚拟机一套。

五、其它要求事项

（1）电子文档必须按如下规范：

实验报告电子文档：**ECOP-学号-XX.PDF**，其中 XX：代表第几次实验，如 01、02…;

必须注意：“ECOP”与“学号”与“XX”用“-”连接，而不用“_”。

其它相关的设计文档：**ECOP-学号-XX.RAR**，全部打包在该文件中。

以上两个文件独立存放，但必须同时提交。如果不交这两个文档，本次实验没成绩。

（2）实验必须在规定时间内完成，尽量在正常上课时间内接受提问性实验检查，而且每位同学都必须通过检查这一关，这样本次实验才算完成。不在规定时间内完成的实验，扣分。

（3）实验报告必须按模板要求严格执行，不合规范、书写简单及随便表达文意等等，扣分。提交的报告中需对自己的心路历程和实验步骤进行相应描述，不能空有答案。必须注意：如果发现实验报告有抄的嫌疑，扣分是很重的！

（4）必须注意：没有通过检查的实验，如果只提交相关电子文档，该次实验不计入成绩。

（5）炸弹采用联网验证答案，请在完成过程中全程连接校园网，并不定期查看排行榜上自己所对应的炸弹的拆解情况，确保服务器准备记录了你的拆弹结果。注：排行榜仅作核对自己的完成情况是否被服务器记录之用，不作为评分依据。