二进制炸弹

任务简介

- 二进制炸弹是一个由C语言编译成的Linux可执行文件，包含若干个phase，每个phase会从输入文件中读入一行字符串，所有字符串正确则炸弹被排除，否则.....
  - 任务是通过对可执行文件的逆向来解出这些字符串
  - 用以下命令来运行，其中solution.txt是输入文件
    - ./bomb solution.txt
    - solution.txt中每行对应一个phase

实施步骤

- bombs文件夹中包含47个bomb代码包（.tar）
- 根据“学号后三位%47”领取自己的代码包
  - 例：学号后三位004，领取 bomb4.tar
  - 例：学号后三位047，领取 bomb0.tar
- 解压后有如下文件
  - bomb　二进制可执行文件，即二进制炸弹
  - bomb.c　bomb的源文件，用于辅助理解bomb的执行过程

实施步骤

- 查看bomb.c可知程序利用phase_*函数（*为1~6） 检查输入字符串是否合法，不合法就引爆炸弹。我们的任务就是逆向出每个phase的检查规则，构造出合法字符串。
  - 当然，bomb.c中没有展示phase_*的源码
- 逆向方法
  - gdb
  - 反汇编（请查阅 gcc –S 命令）
- 以上2种方法可以联合使用

参考索引

- GDB参考
  - 课本3.12（第二版3.11）节，“现实生活：使用GDB调试器”
  - 更详细一些的资料：

  http://heather.cs.ucdavis.edu/~matloff/UnixAndC/CLanguage/Debug.html
  - 官方网站

  http://www.gnu.org/software/gdb/

参考索引

- X86指令手册：
  http://csapp.cs.cmu.edu/public/students.html
- GNU 汇编语法参考
  - Wiki的简明介绍

    https://en.wikibooks.org/wiki/X86_Assembly/GAS_Syntax
  - 手册

    https://sourceware.org/binutils/docs/as/
- 栈的结构：课本3.7节“过程”