15-213 秋季 20xx

实验室任务 L2:拆除二元炸弹 分配时间:9 月 13 日,截止日期:9 月 22 日星期五

Harry Bovik (bovik@cs.cmu.edu) 是该实验室的负责人。

1 简介

邪恶的邪恶博士在我们班的机器上植入了大量的“二元炸弹”。二元炸弹是由一系列阶段组成的程序。每个阶段都希望您在标准输入上键入特定的字符串。
如果您键入正确的字符串,则该阶段被解除,炸弹进入下一个阶段。否则,炸弹会通过打印“BOOM!!!”而爆炸然后终止。当每个阶段都被拆除时,炸弹就会被拆除。

炸弹太多了,我们要处理,所以我们给每个学生一个炸弹来化解。你的任务,你别无选择,只能接受,就是在截止日期前拆除你的炸弹。祝你好运,欢迎来到拆弹队!

第 1 步:拿到炸弹

您可以通过将 Web 浏览器指向以下位置来获取炸弹:

http://$Bomblab::SERVER_NAME:$Bomblab::REQUESTD_PORT/

这将显示一个二进制炸弹请求表供您填写。输入您的用户名和电子邮件地址,然后点击提交按钮。服务器将构建您的炸弹并将其以名为 bombk.tar 的 tar 文件返回到您的浏览器,其中 k 是您的炸弹的唯一编号。

将 bombk.tar 文件保存到您计划在其中进行工作的（受保护的)目录。然后给出命令:tar -xvf bombk.tar。这将创建一个名为 ./bombk 的目录,其中包含以下文件:

- 自述文件:识别炸弹及其所有者。
- 炸弹:可执行二进制炸弹。

- bomb.c:包含炸弹主程序的源文件和来自邪恶博士的友好问候。
- writeup.{pdf,ps}:实验室记录。

如果出于某种原因您请求多个炸弹,这不是问题。选择一个炸弹来处理并删除其余的。

第 2 步:拆除炸弹

你在这个实验室的工作是化解你的炸弹。

您必须在其中一台课堂机器上完成作业。事实上,有传言说邪恶博士真的是邪恶的,如果跑到别处,炸弹总是会爆炸。炸弹中还内置了其他几种防篡改设备,或者我们听到了。

您可以使用许多工具来帮助您化解炸弹。请查看提示部分以获取一些提示和想法。最好的方法是使用你最喜欢的调试器来单步调试反汇编的二进制文件。

每次你的炸弹爆炸时,它都会通知炸弹实验室服务器,你会在实验室的最终分数中失去 1/2 分(最多 20 分)。所以引爆炸弹是有后果的。你一定要小心!

前四个阶段各得 10 分。第 5 阶段和第 6 阶段稍微困难一些,因此它们各得 15 分。所以你能得到的最高分是70分。

尽管阶段变得越来越难以化解,但您在从一个阶段移动到另一个阶段时获得的专业知识应该可以抵消这一困难。但是,即使是最优秀的学生,最后阶段也会挑战,所以请不要等到最后一分钟才开始。

炸弹忽略空白输入行。例如,如果您使用命令行参数运行炸弹,

```
linux> ./bomb psol.txt
```

然后它将从 psol.txt 读取输入行,直到到达 EOF(文件结尾),然后切换到标准输入。在一个虚弱的时刻,Dr. Evil 添加了这个功能,这样你就不必为你已经化解的阶段重新输入解决方案了。

为避免意外引爆炸弹,您需要学习如何单步执行汇编代码以及如何设置断点。您还需要学习如何检查寄存器和内存状态。做这个实验的好处之一是你会非常擅长使用调试器。这是一项至关重要的技能,将在您的职业生涯中为您带来丰厚的回报。

后勤

这是一个单独的项目。所有的handins都是电子的。澄清和更正将张贴在课程留言板上。

提交

没有明确的handin。炸弹会在您处理它时自动通知您的教练您的进度。你可以通过查看班级记分牌来跟踪你的表现：

http://$Bomblab::SERVER_NAME:$Bomblab::REQUESTD_PORT/scoreboard

该网页会不断更新以显示每个炸弹的进度。

提示（请阅读此内容！）

有很多方法可以化解你的炸弹。您可以在不运行程序的情况下详细检查它，并弄清楚它到底做了什么。这是一种有用的技术，但并不总是那么容易做到。您也可以在调试器下运行它，逐步观察它的工作，并使用这些信息来化解它。这可能是化解它的最快方法。

我们确实提出了一个要求，请不要使用暴力！您可以编写一个程序，尝试所有可能的键来找到正确的键。但这并不好，原因有以下几个：

- 每次您猜错并且炸弹爆炸时，您将损失 1/2 分（最多 20 分）。
- 每次您猜错时，都会向bomblab 服务器发送一条消息。您可以很快用这些消息使网络饱和，并导致系统管理员撤销您的计算机访问权限。
- 我们没有告诉你字符串有多长，也没有告诉你其中有哪些字符。即使您做出（不正确的）假设，即它们的长度都小于 80 个字符并且仅包含字母，那么每个阶段您将有2680 次猜测。这将需要很长时间才能运行，并且在作业到期之前您不会得到答案。

有许多工具旨在帮助您弄清楚程序是如何工作的，以及当它们不工作时出了什么问题。这里列出了一些您可能会发现对分析炸弹有用的工具，以及如何使用它们的提示。

- gdb

  GNU 调试器，这是一个命令行调试器工具，几乎可以在每个平台上使用。您可以逐行跟踪程序，检查内存和寄存器，查看源代码和汇编代码（我们不会为您提供大部分炸弹的源代码），设置断点，设置内存观察点，然后编写脚本。

  CS:APP 网站

  http://csapp.cs.cmu.edu/public/students.html

  有一个非常方便的单页 gdb 摘要，您可以打印出来并用作参考。以下是使用 gdb 的其他一些技巧。

–为了防止每次输入错误时炸弹爆炸,您需要学习如何设置断点。

–对于在线文档,在 gdb 命令提示符下键入“help”,或在 Unix 提示符下键入“man gdb”或“info gdb”。有些人也喜欢在 gdb-mode 下运行 gdb
emacs。

- objdump -t

  这将打印出炸弹的符号表。符号表包括炸弹中所有函数和全局变量的名称、炸弹调用的所有函数的名称及其地址。您可以通过查看函数名称来学习一些东西!

- objdump -d

  使用它来反汇编炸弹中的所有代码。您也可以只查看各个功能。
  阅读汇编代码可以告诉你炸弹是如何工作的。

  尽管 objdump -d 为您提供了很多信息,但它并没有告诉您全部情况。对系统级函数的调用以一种神秘的形式显示。例如,可能会出现对 sscanf 的调用
  作为:

  8048c36:e8 99 fc ff ff 调用 80488d4 <_init+0x1a0>

  要确定该调用是对 sscanf 的调用,您需要在 gdb 中进行反汇编。

- 字符串

  该实用程序将在您的炸弹中显示可打印的字符串。

寻找特定的工具?文档呢?不要忘记,命令 apropos、man 和 info 是你的朋友。特别是， man ascii 可能会派上用场。 info gas 会给你比你想知道的更多关于 GNU 汇编器的信息。此外,网络也可能是信息的宝库。如果您遇到困难,请随时向您的教练寻求帮助。