COMP 201 - Fall 2022
Assignment 4 - Defusing a Binary Bomb

Assigned: 12th December 2022, Due: 25 December 2022 23:59

Burak Can Biner (`bbiner21@ku.edu.tr`) is the lead person for this assignment.

1. Introduction:

The nefarious *Dr. Evil* has planted a slew of "binary bombs" on our class machines. A binary bomb is a program that consists of a sequence of phases. Each phase expects you to type a particular string on `stdin`. If you type the correct string, then the phase is *defused* and the bomb proceeds to the next phase. Otherwise, the bomb *explodes* by printing `"BOOM!!!"` and then terminating. The bomb is defused when every phase has been defused.

There are too many bombs for us to deal with, so we are giving each student a bomb to defuse. Your mission, which you have no choice but to accept, is to defuse your bomb before the due date. We are eagerly looking at the scoreboard that reflects how you are moving forward with the bombs. In the followings, you can find how to obtain your bomb and check the scoreboard.

Good luck, and welcome to the bomb squad!

1.1 Step 1: Get Your Bomb

You can click here and type your KUNET ID and email you can obtain your bomb package. Please don't forget that your KUNET ID is your your email without @ku.edu.tr. Note that the webserver may be a bit slow, so please wait a little before hitting the submit button again.

- bomb: The executable binary bomb file.

- bomb.c: Source file with the bomb's main routine.

- README: Identifies the bomb and its owners.

We don't use GitHub classroom for this assignment and instead you will obtain your bomb from the provided URL. For copying file from your own machine to the LinuxPool accounts you may want to look at scp (copying files and folders over ssh connection) command.

WARNING: Don't forget that you can only work on your bombs on the LinuxPool machines. You can click this link to check the progress for each bomb. Note that this scoreboard timezone is not same with Turkey timezone.

1.2 Step 2: Defuse Your Bomb

You can use many tools to help you defuse your bomb. Please look at the hints section for some tips and ideas. The best way is to use your favorite debugger to step through the disassembled binary. Each time your bomb explodes it notifies the bomblab server, and you lose 1/2 point (up to a max of 20 points) in the

final score for the lab. So there are consequences to exploding the bomb. Although phases get progressively harder to defuse, the expertise you gain as you move from phase to phase should offset this difficulty. However, the last phase will challenge even the best students, so please don't wait until the last minute to start.

The bomb ignores blank input lines. If you run your bomb with a command line argument, for example,

```
$ ./bomb psol.txt
```

then it will read the input lines from psol.txt until it reaches EOF (end of file), and then switch over to stdin. In a moment of weakness, Dr. Evil added this feature so you don't have to keep retyping the solutions to phases you have already defused. To avoid accidentally detonating the bomb, you will need to learn how to single-step through the assembly code and how to set breakpoints. You will also need to learn how to inspect both the registers and the memory states.

2. Hints

2.1 $gdb FILENAME

To keep the bomb from blowing up every time you type in a wrong input, you'll want to learn how to set breakpoints and use gdb. Here is a short video that shows how to use gdb and commands to go through a disassembled code. The basics are the same as what you learned during your **gdb** lab. There are treasures commands in the mentioned video, so watch it carefully.

2.2 $objdump -t FILENAME

This will print out the bomb's symbol table. The symbol table includes the names of all functions and global variables in the bomb, the names of all the functions the bomb calls, and their addresses. You may learn something by looking at the function names!

2.3 $objdump -d FILENAME

Use this to disassemble all of the code in the bomb. You can also just look at individual functions. Reading the assembler code can tell you how the bomb works.

Although **objdump -d** gives you a lot of information, it doesn't tell you the whole story. Calls to system-level functions are displayed in a cryptic form. For example, a call to **sscanf** might appear as:

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

To determine that the call was to **sscanf**, you would need to disassemble within **gdb**.

2.4 $strings FILENAME

This utility will display the printable strings in your bomb maybe something can be found there.

3. Submission

We use Blackboard for this assignment ONLY! You are expected to submit a psol.txt file in which each line correspond to a different bomb phase. **In addition, you should submit a report.pdf file explaining critical points and strategies you followed to solve each phase.**

4. Oral Assessment

Important Note: We plan to ask randomly selected 10% of students to explain their approach verbally after the assignments are graded. And one may lose full credit if he or she fails from this oral part.

5. How to use linuxpool.ku.edu.tr linux servers

  I Connect to KU VPN (If you are connected to the KU network, you can skip this step.)
    See for details: https://confluence.ku.edu.tr/kuhelp/ithelp/it-services/network-and-wireless/vpn-access

  II Connect to linuxpool.ku.edu.tr server using SSH (Replace USER with your Koç University username):
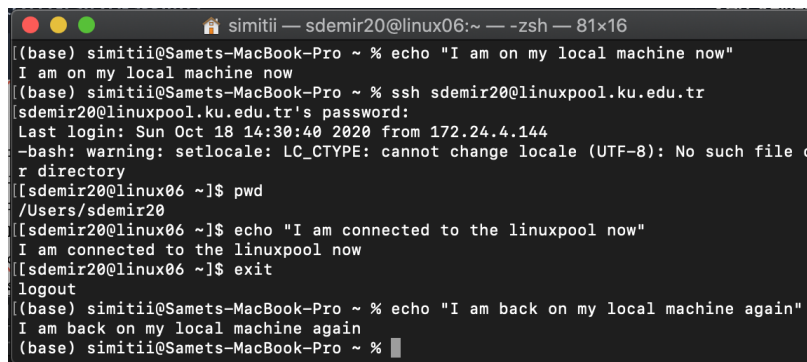
```
ssh USER@linuxpool.ku.edu.tr
```

    (It will ask your password, type your Koç University password.)

 III When you are finished with your work, you can disconnect by typing: `$ exit`

    Your connection to the server may drop sometimes. In that case, you need to reconnect.

    We advise you to watch the following video about the usage of SSH, which is used to connect remote servers, and SCP, which is used to transfer files between remote servers and your local machine: https://www.youtube.com/watch?v=rm6pewTcSro

Figure 1: How to connect and disconnect using SSH

6. Academic Integrity

All work on assignments must be done individually unless stated otherwise. You are encouraged to discuss the given assignments with your classmates, but these discussions should be carried out in an abstract way. That is, discussions related to a particular solution to a specific problem (either in actual code or in the pseudocode) will not be tolerated. In short, turning in someone else's work, in whole or in part, as your own will be considered as a violation of academic integrity. Please note that the former condition also holds for the web material as everything on the web has been written by someone else. See Koç University - Student Code of Conduct.

7. Late Submission Policy

You may use up to 7 grace days (in total) over the course of the semester for the assignments. That is, you can submit your solutions without any penalty if you have free grace days left. Any additional unapproved late submission will be punished (1 day late: 20% off, 2 days late: 40% off) and **no submission after 2 days will be accepted.**

8. Acknowledgment

This assignment is adapted from CMU CS15-213 course contents.