Portland State University

CS201: Computer Systems Programming

Homework 4

Due: March 4th, 2020 at 11:59 pm

About MetaCTF

- The concept of "Capture the Flag" may be new to you. This is similar to the real-world game of the same name, but with much simpler rules. The purpose of the MetaCTF is to give you practice with the concepts and techniques discussed in class. Each MetaCTF problem has a specific learning objective. Some will be as simple as familiarizing you with the GNU debugger, gdb, and others will, for example, help you to understand the layout and purpose of the process stack. A secondary goal is to prepare you for security CTFs that you will encounter, should you go on to study computer security here at PSU or elsewhere. Since the CTFs will be in the form of executable binaries, there is an emphasis on assembly code and executable file formats. On Linux, the default executable format is called "ELF", which stands for "Executable and Linkable Format".

- One of the reasons to learn assembly code and binary executable formats is to reverse-engineer unknown, potentially malicious, software. MetaCTF is a capture-the-flag game that is intended to teach concepts in this course in a more interesting and fun manner. It consists of a set of Linux ELF binaries that are unique to each student. For each binary, you are seeking an input that will force the binary to print out "Good Job". Instructions are given for each binary when they are run in order to guide you. You are to submit your "winning" inputs that solve your binaries via the web site as shown below.

- Note that the concepts and techniques for each student are the same, only the solution will differ. As students may discuss techniques with each other, the CTFs have the side benefit of encouraging students to collaborate on knowledge and understanding of the techniques as well as the concepts discussed in class.

Your binaries

- Each user has a set of binaries (and some source files) located at the following URL
    - https://cs201.oregonctf.org
    - To log in, **your username is your Odin ID and your password is the last 4 digits of your PSU student ID**
    - Please let me or the TA know immediately if you cannot log in.

- Once logged in, there will be two options: Download and Solve
    - Use the Download link to get a zip file of your binaries.
    - Use the Solve link to submit individual answers to each binary.

- **If you are unable to run the executables**, you may need to execute this command in the directory where you have the executables: "chmod +x *" .

Example Problem (Ch1_Ltrace)

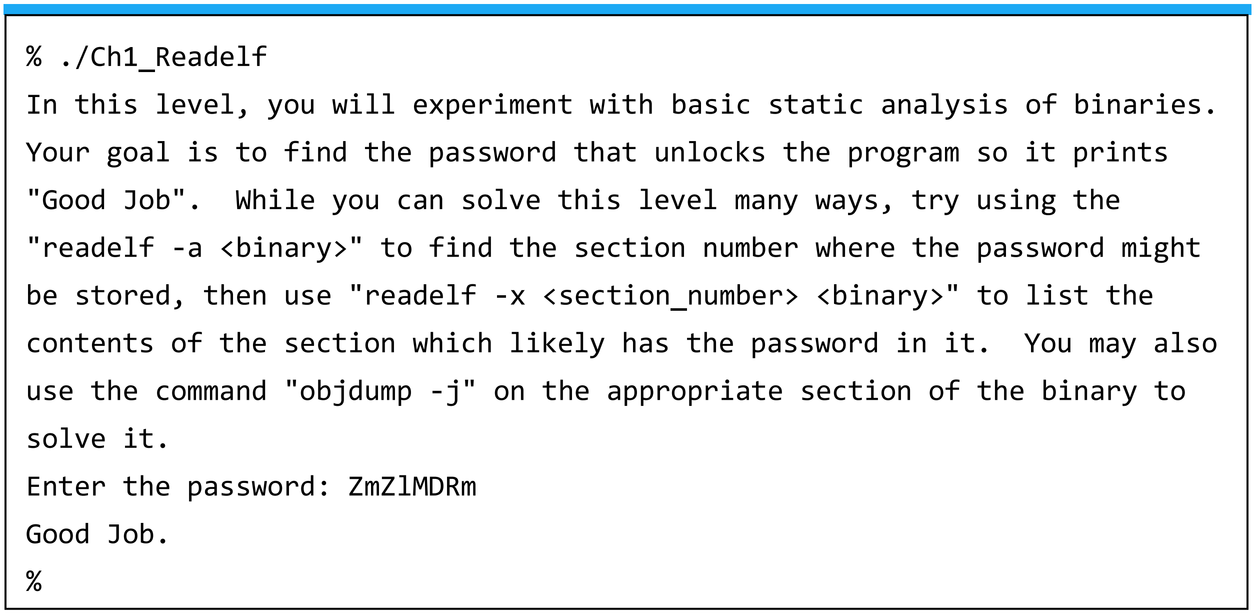
The following example shows how to solve Ch1_Ltrace MetaCTF problem:

1. Downloading and extract the binaries for the chapter you are trying to solve. Executing the `ls` command should show different binaries for each exercise in the chapter

```
% ls

Ch1_Ltrace
Ch1_Readelf
```

2. Run the desired executable for the problem you are trying to solve.



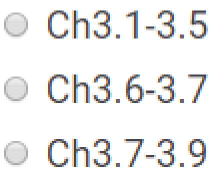
```
% ./Ch1_Readelf
In this level, you will experiment with basic static analysis of binaries.
Your goal is to find the password that unlocks the program so it prints
"Good Job".  While you can solve this level many ways, try using the
"readelf -a <binary>" to find the section number where the password might
be stored, then use "readelf -x <section_number> <binary>" to list the
contents of the section which likely has the password in it.  You may also
use the command "objdump -j" on the appropriate section of the binary to
solve it.
Enter the password: ZmZlMDRm
Good Job.
%
```

3. Use the MetaCTF walkthrough included as part of this package provides additional guidance on how to solve that particular exercise

4. Use the password provided by the exercise to answer the questions for that particular exercise in the PSU CTF website

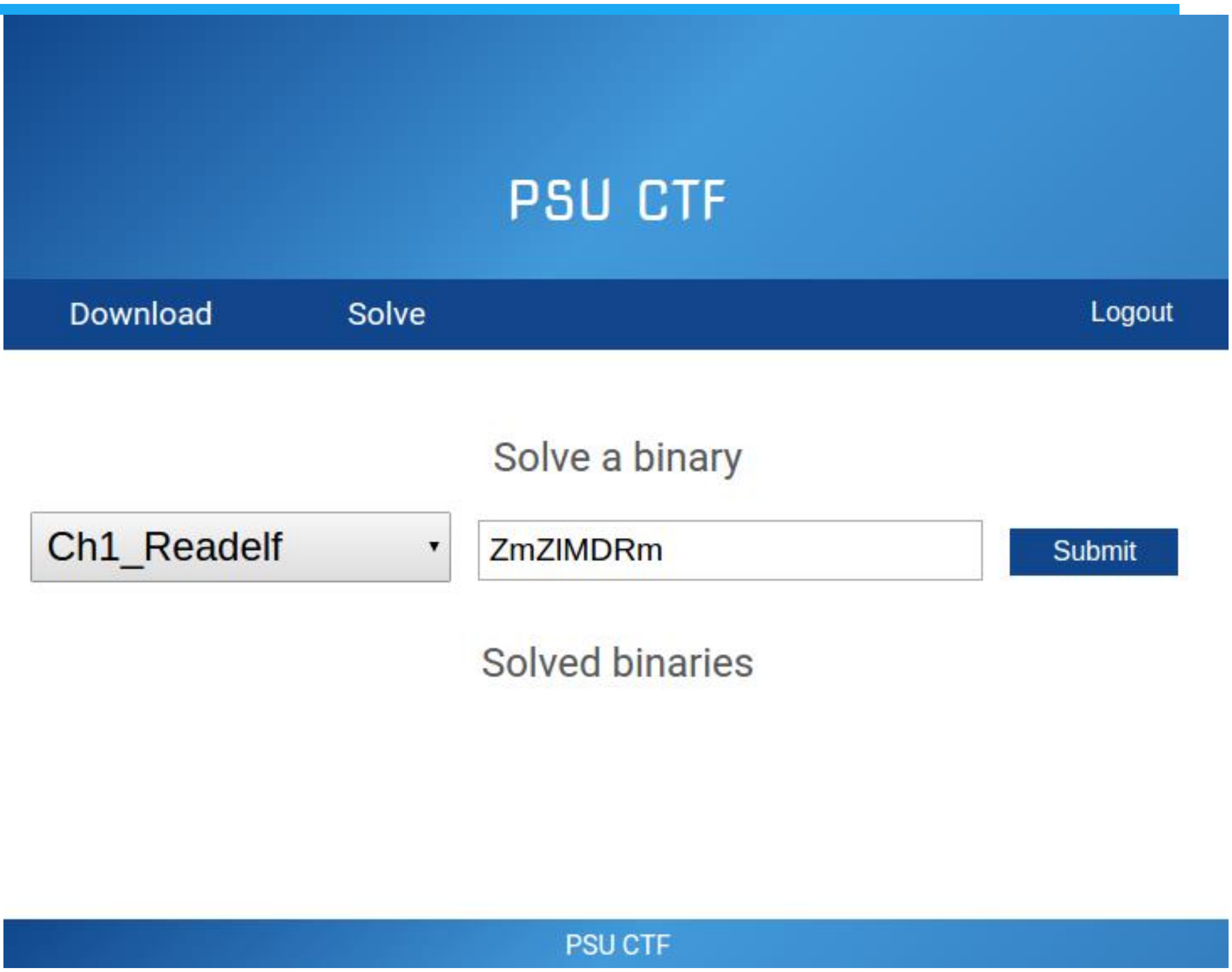
Assignment Instructions

For this assignment we will solve **Chapter 3 (3.1 - 3.9) of the MetaCTF set only**.
Please note that you only need to download the following sets:



○ Ch3.1-3.5
○ Ch3.6-3.7
○ Ch3.7-3.9

Use the PSU CTF Website to submit your solutions.



Solve a binary

Ch1_Readelf ▾

ZmZlMDRm

Solved binaries

5.