CSC 495/583 Fall 2017 Lab 3

Dr. Si Chen

Due on: 11/14/2017

Heap Overflow

Purpose: To practice exploiting a very simple heap overflow vulnerability. This one is easy to exploit because there's a pointer in the heap that is used for a function call. That makes a heap overflow as simple as a stack overflow targeting EIP.

What You Need: A 32-bit x86 Kali Linux machine, real or virtual. The project was written on Kali 2.

Objectives and Targets

Target: Very Simple Heap Overflow

The provided C code (lab3.c) contains a heap overflow vulnerability. Please write an exploit (by modifying data.txt) to output **"level passed"** on Linux. The high level idea is to overwrite the return address with the address of function winner(). Once the return instruction is executed, this function will be called and output the string.

We have provided you with a virtual machine image for this project, use the latest version of VirtualBox.

Our VM's image link can be found on our course website:
https://www.cs.wcupa.edu/schen/security/
We suggest using wget to ensure that you've downloaded the file correctly -
wget -c https://www.cs.wcupa.edu/schen/security/download/Kali-Linux-2017.2-vbox-i386.ova
Steps:

0) Boot up the VM, the default username is: root and password is: toor

1) Download the provided C code from our course website inside virtual machine, open it with any code editor.

2) Compile the provided C code (which you will be exploiting):

```
gcc lab3.c -o lab3 -m32 -fno-stack-protector -zexecstack -z norelro
```

3) To run this program, create a new file called data.txt and put some string literals in it, and execute lab3 by:

```
./lab3 shellcode
```

4) Provide a screenshot of you exploiting sort.
5) Have fun.

Submission

- The project is due by 11/14/2017. Late submission will not be accepted;
- The assignment should be submitted to D2L directly.
- Your submission should include: A **detailed project report in PDF format** to describe what you have done, including screenshots and code snippets.
- **No copy or cheating is tolerated**. If your work is based on others', please give clear attribution. Otherwise, you **WILL FAIL** this course.