# CSC 472/583 Fall 2020 Lab 4

Dr. Si Chen

## Return-to-libc Attack

The goals of this lab:

- Understanding the concepts of return-to-libc attack
- Exploiting a return-to-libc vulnerability

### Objectives and Targets

#### Target: Launch Return-to-libc Attack

The provided C code (lab4.c) contains a stack buffer overflow vulnerability. Please create a Return-to-libc Attack payload. The high level idea is to overwrite the return address with the address of function *system()* in libc and then pass the string *"/bin/sh"* as argument. You should get a new shell (e.g. executing a linux command '/bin/sh') if success.

You can use our Badger CTF system to complete this project. Please check the video tutorial on our course website ("how to connect to Badger CTF with my own account").

**Steps:**

1) Download the provided C code from our course website.

```
wget https://www.cs.wcupa.edu/schen/ss2020/lab/lab4.c
```

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

```
gcc -m32 -fno-stack-protector -no-pie lab4.c -o lab4
```

3) To run this program, type the following command:

```
./lab4
```

4) When you type a very long list of characters, you will notice lab4 crashes with memory segfault, this is because the return address has been overwritten by your data.

5) Now you can craft your payload using the provided skeleton Python script. Again, your goal is to overwrite the return address with the address of function **system()** in libc and then pass argument **"/bin/sh"** (also available in libc). GDB can be used to find these library addresses and test/debug your exploit. However, it should be noted that your final exploit (i.e., the final version of your Python script) should work outside of GDB. Just running

```
python3 return2libc.py
```

You should get a new shell if success.

6) Provide a screenshot of you exploiting sort.

7) Have fun.

Deliverables: the Python script you crafted and a screenshot of the exploit. The structure of your payload (Python code) and the screenshot should be put into the PDF file.

# More...

Please check lecture video

# Submission

- The lab due date is available on our course website. 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.