Lab Project #3: Attack Lab — Buffer Overflow Attacks

Professor Hugh C. Lauer
CS-2011, Machine Organization and Assembly Language

Today

- **Attack Lab assignment**
  - Due Saturday, December 2, 2017 (6:00 PM)

- **Buffer Overflow**
  - Vulnerability
  - Protection

Attack Lab— Objective and Approach

- **To gain a deeper understanding of x86_64 calling conventions and stack structure …**

- **… by devising a series of buffer overflow attacks on two executable files called *targets***
  - I.e., make information show up in places in memory where it ordinarily would not show up

Warning!
What you are about to do would be highly illegal if carried out against any system outside of this class project!

Obtaining Attack Lab

- **Similar to Bomblab**
  - Contact server at

    http://cs2011.cs.wpi.edu:15513
  - Note "socket" number, similar to but different from Bomblab
- **Will turn on Attacklab Sunday, November 19**
  - After turning off Bomblab!
- **Downloads a tar file with a lot of stuff in it**
  - Two "vulnerable" executables
  - Other supporting stuff

Obtaining Attack Lab (continued)

- **Okay to download anytime!**
- **Let us know of difficulties**
- **We are able to add your machine name to list of authorized machines**
  - If you are not already authorized from Bomblab, please let us know!

This Project

- **Five levels of attack on two separate executable programs**

- **For each one, you must:–**
  - Write some C code based on your disassembly of the **`ctarget`** or **`rtarget`** binary
  - Generate the corresponding x86_64 assembly code (**`gcc -s`**) and create an *exploit string*
  - Convert this into a raw string to pass into **`stdin`** (i.e., **`hex2raw`**)
  - Successfully plant your “cookie” or other information in your target

- **Make *your* cookie show up some place where it should not be!**
  - Solution is different for each student (i.e., each cookie)

Attack Lab server

- **When successful, submit the exploit string to the online grading server**

- **http://cs2011.cs.wpi.edu:15513/**
  - Will build and deliver a new target
  - Please don't do it at Recitation time — overloads the server
  - All Attack targets are different

Scoreboard

- http://cs2011.cs.wpi.edu:15513/scoreboard

- **Indexed by target number!**
  - Similar to Bomb #

Working on Attack Lab

- ***Must be submitted from virtual machine of this course***
  - *Or other authorized machine*
- **No penalty for mistakes**

Due date:–December 2, 2017, 6:00 PM

Questions?

From second day of this course:–

```
double fun(int i)
{
  volatile double d[1] = {3.14};
  volatile long int a[2];
  a[i] = 1073741824; /* Possibly out of bounds */
  return d[0];
}
```

```
fun(0)  →    3.14
fun(1)  →    3.14
fun(2)  →    3.1399998664856
fun(3)  →    2.00000061035156
fun(4)  →    3.14, then segmentation fault
```

- **Result is architecture specific**

Tantalizing preview of approach to AttackLab

Overflow a buffer with some carefully planned data to cause program to do something different!

Questions?