Lecture 1 Notes

* Every lecture-based challenges should be completed. Most of these will be gone over during the weeks that they’re due.
* Every grad student will have to create they’re own challenges
* All the due dates are under the schedule
* The discussion portion of the grade will just be graded for just showing up to class.

**Stack0 Overview:**

-setuid gives permissions to the binary for what can and can’t be read

-declaring a variable as volatile tells the compiler to not optimize it

-gets() is the function that can be exploited since it does not perform boundary checks

**Virtual Memory:**

A virtual address space for a process

It’s an abstraction that the OS provides to each process. Each process does not see physical memory, they see the virtual address space.

It’s basically a giant block of memory that starts at 0x0 to 2^64 on a 64bit machine

Stores the: Text/code, Heap, Stack, Data (Dynamically linked libraries)

* Heap is programmer managed
* Stack is used the compiler for allocations. The only reason functions work is because of the stack. It adds a Stack Frame to the top of the stack. Stack frames can include control information, local variables, and the arguments to the function.

**GDB(Gef):**

* Info locals prints out the values of all local variables
* P &(variable) will give the exact memory address of it
* Can do p (addr) – (addr) to find the exact amount of memory between them (ret addr)
* p/d (addr) will give out the decimal amount
* x/s (examine string) &(address)

**Pwntools:**

* Useful for making the process of going through the exploit process easier using python.