**Lecture 3 Notes**

**First part of class spent talking about web security topics about passwords we covered in 558**

**(To solve Muffin Fan Club)**

**Stack3 Challenge:**

* No obvious variable function pointer to overflow memory inside the main() function
* **Ni = next instruction n = next in GDB**
* **Virtual address space holds the text, data, heap, stack, shared libs and kernel info**
* Function memory location is compiled into the text section and referenced by the stack frame later when it’s called
* **Calling Convention:** Dictates how one function calls another. In stack3, the calling convention state that arguments will be passed via registers.
* **RBP register points to the start/top of the stack frame**
* **sub assembly call allocates memory for local variables in the new stack frame by changing the stack pointer location with an offset.**
* **Register spilling is where registers save values to the stack before overwriting it.**
* **All addresses are 8 bytes long. GDB may sometimes not display all addresses like that.**
* **Pwntools will automatically make sure its 4 or 8 bytes when using p32 or p64()**
* **Control is transferred to another function by setting a register value to hold the address in the RIP register. (instruction pointer)**