

# “Smashing the Stack for Fun and Profit”

(aka Project 1)

# A Crash Course in x86

- General-purpose registers:
  - EAX, EBX, ECX, EDX, ESI, EDI, ...
- Stack registers:
  - ESP – Current stack pointer. Grows down.
  - EBP – Stack frame pointer.  
Used to address locals, arguments, etc.
- EIP – Instruction Pointer/Program Counter

# A Crash Course in x86

- Stack manipulation instructions:
  - PUSH arg:  $ESP - 4 \rightarrow ESP$ ,  $arg \rightarrow @ESP$
  - POP arg:  $@ESP \rightarrow arg$ ,  $ESP + 4 \rightarrow ESP$
- Function call instructions:
  - CALL addr: PUSH EIP, JMP addr
  - RET: POP EIP
  - LEAVE: EBP  $\rightarrow$  ESP, POP EBP

# A Crash Course in GDB

- info reg: Dump registers
- disassemble (Only for the current function)
- run: (re)start program
- continue
- break: set a breakpoint:
  - b foo
  - b foo+10
  - b \*0x080fcde8
- step, stepi

# A Crash Course in GDB

- x: Examine
  - x buf
  - x \$esp
  - x 0x0804face
  - x/20i main
  - x/40x buf
  - ...

**Let's h4x0r!!!11**