Homework 10: Buffer Overflow

(a) Also called stack smashing, stack buffer overflow is dangerous because it can overwrite ___ pointer.

Return Address

(b) Shell code executed as a result of stack buffer overflow attack is executed at whose privilege?

The hijacked program's. This often results in an elevation of privilege for the attacker, which is bad.

(c) Defenses against buffer overflow can be carried out at compile-time or at run-time. Name one defense in each category.

Acceptable answers include...

Compile-time:

- Use high-level programming languages that don't interactive with low-level details.
- Use updated and safe libraries.
- Enable stack protection (checking for stack corruption, copying return address pointer, etc.) This is often done by default today.

Run-time:

- Distinguish executable address space from non-executable ones.
- Randomize address (so that the attacker can't predict what to overwrite the return address with)

(d) **Bonus:** why can't shell code have null bytes? How do people obtain a zero value in shell code?

Because shellcode is often overwriting strings, which ends at a null byte. By xor'ing a value with itself.