Buffer Overflow Review

What is buffer overflow? How is it exploited? Describe the parts of the stack and where they are.

* Page 5, 6

What is the goal of a BO attack and what does it involve? How do you place code in the address space?

* Page 11, 12, 13, 14

How do you jump to the attacker code? What is a buffer? What are the 3 generations of exploits for BO?

* Page 15, 16

What is stack smashing? Describe the sections of the stack.

* Page 18, 19

What happens when memory outside a buffer is accessed? What is a stack frame? What happens when strcpy is called? What about when the buffer is overstuffed?

* Page 20, 21, 22, 23

How is the attack code executed? What is the attack procedure?

* Page 24, 26

What are some issues that must be overcome to exploit the BO? How would you get the buffer address? What is a NOP sled?

* Page 27, 28, 29

What are the unsafe C functions? How does a non-executable buffer stop BO? What about address randomization?

* Page 40, 43, 46, 47

How does Array bounds checking stop BO? What is a stack guard? How is it implemented?

* Page 48, 50, 51, 52