# Quiz: Buffer Overflow Intro

10 Questions, Answers follow the “About” page.

Author: Jedidiah R. Crandall, crandaj@erau.edu

This Document was Funded by the National Science Foundation
Federal Cyber Service Scholarship For Service Program:
Grant No. 0113627

Distributed July 2002

Embry-Riddle Aeronautical University • Prescott, Arizona • USA

# 1. Which of these statements about the buffer overflow problem are correct?

A. The buffer overflow problem is partly caused by the way the C language handles memory management
B. The buffer overflow problem is partly caused by C programmers not handling their own memory management properly by checking boundaries of buffers
C. All buffer overflows are simple programmer errors that are easily spotted
D. Because of the complexity of the problem, buffer overflows may be overlooked by the most seasoned programmers

# 2. What can make a buffer overflow a security problem?

A. Only when the attacker is able to hijack the execution of the program
B. Only when the buffer overflow is between two computers on a network
C. When security-sensitive data is overwritten
D. When data that is critical to the execution of the program is overwritten causing the program to crash

# 3. What can be stored using a single byte?

A. Integers from 0 to 255
B. ASCII Characters
C. Integers from 0 to 4294967296
D. Boolean variables (i.e. 1 = Okay to access the file and 0 = Not okay to access the file)

# 4. What can be stored in a computer's memory?

A. Data such as integers and ASCII charcaters
B. Instructions for a program that the computer is running
C. A pointer to the address of some other data in memory
D. A pointer to the address of some more instructions in memory

# 5. What typically happens when a buffer is overflowed?

A. The memory space that comes after the buffer holds the extra data as well as keeping the data that it contained before
B. Whatever is in the memory space that comes after the buffer is overwritten
C. The memory chip in the computer gets too big and explodes
D. Electrons fall out of the memory chip and start a fire

# 6. What does it mean for a program to jump?

A. The whole program gets moved from one place in memory to another
B. The program starts executing instructions at a different location in memory instead of moving on to the next one
C. The program jumps outside the bounds of a buffer to store data
D. Jump is only used to refer to when a program starts executing instructions somewhere where it's not supposed to

# 7. Which of these is a good example of subroutines?

A. A stack of trays in the cafeteria
B. Pouring five gallons of water into a four-gallon bucket
C. An IRS form which has boxes where you write things like adjusted income, exemptions, wages, tips, etc.
D. A program for brushing your teeth is broken up into subroutines: getting out your toothbrush and toothpaste, putting toothpaste on the toothbrush, brushing your molars on the right, brushing your molars on the left, …

# 8. What does a typical C program usually use stacks for?

A. Temporary storage of variables
B. For storing the computer-level instructions of a subroutine while the subroutine is being executed
C. Keeping track of where it was within subroutines that called other subroutines so it knows where to resume
D. For preventing buffer overflows

# 9. What prevents a typical computer from jumping and starting to execute data instead of instructions?

A. Nothing
B. Buffers are used to separate instructions and data
C. The computer can always tell the difference based on what is in the memory space
D. Data and instructions are stored in separate memory spaces on most modern computers

# 10. What operations are permissible on a stack?

A. Inserting things in the middle
B. Taking things out of the middle
C. Taking things off the bottom
D. Putting things on the bottom
E. Putting things on the top
F. Taking things off the top

# About this Project

This presentation is part of a larger package of materials on buffer overflow vulnerabilities, defenses, and software practices.
For more information, go to: http://nsfsecurity.pr.erau.edu

Also available are:

- Demonstrations of how buffer overflows occur (Java applets)
- PowerPoint lecture-style presentations on an introduction to buffer overflows, preventing buffer overflows (for C programmers), and a case study of Code Red
- Checklists and Points to Remember for C Programmers
- An interactive module and quiz set with alternative paths for journalists/analysts and IT managers as well as programmers and testers
- A scavenger hunt on implications of the buffer overflow vulnerability

Please complete a feedback form at http://nsfsecurity.pr.erau.edu/feedback.html to tell us how you used this material and to offer suggestions for improvements.

# Answers

1. A,B,D
2. C,D
3. A,B,D
4. A,B,C,D
5. B
6. B
7. D
8. A,C
9. A
10. E,F