CS4402 Discussion Assignment 3

**Question 1**: Assume that you are the instructor.  Prepare a short written lecture that describes Big Endian, Little Endian and the advantages and disadvantages of each approach.  
  
**Question 2:** In this unit, we discussed integer data types and integer math.  One of the concepts that we learned about was an overflow.  As part of your response describe what an overflow is an how it occurs.  Conduct research on the internet and based upon your research, describe some of the problems that overflow causes there are many but one area that you should research (among others) is the concept of integer based security attacks.s

**Question1**

According to Hennessey and Patterson (2019), Big Endian and Little Endian differ in the arrangement of bytes within multi-byte data values.

**Big Endian:**

- Ordering: MSB is at the lowest memory address, followed by decreasing order of significance.

- **Advantages**:

- Human Readability: MSB at the start improves human understanding.

- Network Protocols: Many protocols, including IP, use Big Endian for consistent data representation.

- **Disadvantages**:

- Performance: Data conversion between Big Endian and Little Endian can be time and resource-intensive.

- Incompatibility: Some CPUs using Little Endian can lead to performance issues.

**Little Endian:**

- Ordering: LSB is at the lowest memory address, followed by increasing order of significance.

- **Advantages**:

- Performance: Efficient for modern CPUs, reducing byte-swapping operations.

- Consistency: Simplifies data manipulation within a CPU.

- **Disadvantages**:

- Human Readability: Less intuitive for humans due to LSB placement.

- Network Communication: When communicating with Big Endian systems, byte swapping may be required."

**Question2**

**Introduction**

Integer mathematics is crucial in computer programming, allowing manipulation of whole numbers within memory. However, it can lead to issues like overflow when results exceed data type capacity. This discussion explores integer math, overflow, and their relevance to security attacks.

**Integer Math and Overflow**

Integer math involves performing arithmetic operations on whole numbers using fixed-size data types, like int or short. These types have finite ranges; overflow happens when results exceed these bounds. For example, an 8-bit signed integer with a range of -128 to 127, when adding 10 to its maximum value (127 + 10), overflows to -128, highlighting the significance of proper handling (Dowd, McDonald, & Schuh, 2006).

**Relevance to Integer-Based Security Attacks**

Integer overflow vulnerabilities pose grave security threats. Malicious actors exploit them for unauthorized access or executing malicious code. An attacker, by manipulating an integer controlling a data buffer's size, can trigger overflow, overwriting adjacent memory and executing unauthorized code, potentially compromising security (Dowd et al., 2006).

**References**:

Hennessey, J. L., & Patterson, D. A. (2019). Computer Architecture: A Quantitative Approach. Elsevier.

Dowd, M., McDonald, J., & Schuh, J. (2006). *The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities*. Addison-Wesley. ISBN-13: 978-0321444424.