1. Buffer overflow can occur when there is more data written than what a specific buffer can hold. The additional data overwrites other memory locations which could cause unwanted behavior or could even cause a program to crash.

2. Some ways we can prevent buffer overflow are as follows. We could implement preventions that label certain areas of memory as non-executable or executable which helps limit which parts of the program are run when a buffer overflow occurs. We can also implement functions that can check the length of the buffer. Lastly, we can also limit input sizes and implement bounds checking.

3.

4. The code provided that I put into secure_coding.cpp did not check to see if user input exceeded the size of the array. This could potentially cause a buffer overflow so I typed in a check for user input to make sure that it doesn't exceed the max size of the array. After that, the program runs fine.

5. Security Checklist

Security Checklist Line #'s are based off of submitted .cpp file.	
Vulnerability: Buffer Overflow **Course:** CSo	
Task – Check each line of code	Completed
1. Finding Arrays:	✓
1.1 Underline each array declaration — line 14 of file	✓
1.2 For each array, underline all subsequent references — line 35	✓
2. Index Variables *– the range i of legal indices for an array of n elements is 0 <= i < n*	✓
2.1 For each underlined access that uses a variable as an index, write the legal index range next to it. — line 28	✓
2.2 For each index marked in 2.1, underline all occurrences of that variable. — line 29	✓
2.3. Circle any assignments, inputs or operations that may modify these index variables. — line 32	✓
2.4. Mark with a V any array that is indexed by a circled index variable. — line 30	✓
Highlighted areas indicate a buffer overflow vulnerability.	

6. We can prevent the potential buffer by adding a check for the length of the user input to make sure it doesn't exceed the maximum size of the array. This prevents any illegal access that could cause a buffer overflow to occur.
7. The updated code that will be seen with the question 8 file submission was revised to eliminate any potential of buffer overflow.