Hanah Deering

5-3 Activity: Static Code Analysis

CS 405 Secure Coding

Southern New Hampshire University

In this activity, I compared errors that we thrown between Visual Studio and the CppCheck that we downloaded. I prefer the CppCheck because it seems to be more thorough and catches more errors that Visual Studio when building/debugging. Visual Studio only identified 4 errors. However, CppCheck identified three errors, 7 warnings, and a performance issue. This is quite the difference, because if could be the difference of Visual Studio missing one error before the whole system is exposed. Here are the screenshots from visual studio and the results from CppCheck are included in the zipped file.

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

| **LINE#** | **CPP /**  **VS** | **SUMMARY** | **RISK/**  **SEVERITY** | **MITIGATION MEASURE** |
| --- | --- | --- | --- | --- |
| 42 | BOTH | Variable A::x not initialized | WARNING | Initialize variable |
| 50 | CPP | Function can be static | PERFORMANCE | Make function static |
| 52 | BOTH | Exception thrown in function declared not to throw exceptions; Does not return true for all exceptions. | ERROR | Revise code to be more specific to include all exceptions |
| 59 | CPP | Dangling pointer – pointer invalid after end of function; | ERROR | Increase pointer scope rather than local; or add code to reassign pointer prior to end of the function. |
| 64 | CPP | The scope of variable ‘buf’ can be reduced by 1 level | STYLE | Move initialization up one level (prior to the if(x) statement). |
| 66 | BOTH | Array index out of bounds – causes unexpected results. | WARNING | Limit count to between 0 to 9. |
|  | CPP | Unused variable | STYLE | Remove any unused variables from code per best practices. |
| 87 | CPP | Invalid container; resizing (erasing) a vector element (sequence container) reassigns the memory addresses of all successive elements. | ERROR | Use a list instead of a vector |
| 98 | CPP | Method returns an integer instead of a Boolean | STYLE | Revise code to return variable type consistent with method type |
| 109 | CPP | Function parameter ‘tok’ is not used anywhere outside of the function nor is it dereferenced. | WARNING | Dereference pointer ‘tok’ and/or remove the variable/function if it does not serve the rest of the program. |
| 127 | BOTH | Assert statement modifies variable ‘z’. Variable ‘z’ is assigned a constant in the assert function which requires return of a Boolean. | WARNING | Consider using the ‘==’ operator which would return either a true or false result to the assert function. |
| 129 | BOTH | A function is called inside of an assert statement, causing the assertion to fail. Additionally, if my\_function() returned either true or false, it should be set to either 0 or 1 in order to return a Boolean, not an integer. | WARNING | Consider using variable in place of function or returning either a Boolean to replace the integer. |