Daoud Sogoba

CS-405: Secure Coding

Professor Hodde

November 16, 2021

All of the following errors were genreated by CPPCheck as visual studio identified no errors or warnings

(warning) Member variable 'A::x' is not initialized in the constructor. [uninitMemberVarPrivate]

FIX: Variable initializeation in the constructor can prevent run time erros produced by not following this practoce.

(performance,inconclusive) The member function 'MySpecialType::DontThrow' can be made a static function. Making a function static can bring a performance benefit since no 'this' instance is passed to the function. This change should not cause compiler errors but it does not necessarily make sense conceptually. Think about your design and the task of the function first - is it a function that must not access members of class instances? And maybe it is more appropriate to move this function to a unnamed namespace. [functionStatic]

FIX: Making this a static function can save on memeory as no instance of this method will be created.

(error) Exception thrown in function declared not to throw exceptions. [throwInNoexceptFunction]

FIX: The function show be declared as void DontThrow() throws exception rather than void DontThrow() noexcept

(error) Dangerous assignment - the function parameter is assigned the address of a local auto-variable. Local auto-variables are reserved from the stack which is freed when the function ends. So the pointer to a local variable is invalid after the function ends. [autoVariables]

FIX:

(style) The scope of the variable 'buf' can be reduced. Warning: Be careful when fixing this message, especially when there are inner loops. Here is an example where cppcheck will write that the scope for 'i' can be reduced:

FIX: declare interger variable buf inside the for loop that follows it

(style) Variable 'buf[count]' is assigned a value that is never used. [unreadVariable]

FIX: Either make use of this variable or delete it.

(warning) Either the condition 'count==1000' is redundant or the array 'buf[10]' is accessed at index 1000, which is out of bounds. [arrayIndexOutOfBoundsCond]

FIX: Either check count against the actual bounds of buf or change the bounds of buf

(error) Using iterator to local container 'items' that may be invalid. [invalidContainer]

FIX: If the container size has changed the iterator is no longer valid

(style) Non-boolean value returned from function returning bool [returnNonBoolInBooleanFunction]

FIX: Change function return time to integer

(warning) Assignment of function parameter has no effect outside the function. Did you forget dereferencing it? [uselessAssignmentPtrArg]

FIX: Deference the given function param

(warning) Either the condition 'tok' is redundant or there is possible null pointer dereference: tok. [nullPointerRedundantCheck]

FIX: Remove tok or remove the deference

(style) Variable 'tok' is assigned a value that is never used. [unreadVariable]

FIX: Either make use of the tok variable or delete it

(warning) Variable 'z' is modified inside assert statement. Assert statements are removed from release builds so the code inside assert statement is not executed. If the code is needed also in release builds, this is a bug. [assignmentInAssert]

FIX: Assert statements should not exist in non testing enviornments

(warning) Non-pure function: 'my\_function' is called inside assert statement. Assert statements are removed from release builds so the code inside assert statement is not executed. If the code is needed also in release builds, this is a bug. [assertWithSideEffect]

FIX:Assert statements should not exist in non testing enviornments

(warning) Comparison of a boolean expression with an integer other than 0 or 1. [compareBoolExpressionWithInt]

FIX: Use 1 or zero to compare against a boolean

(style) Comparing expression of type 'bool' against value 3. Condition is always false. [compareValueOutOfTypeRangeError]

FIX: Use 1 or zero to compare against a boolean

(style) Local variable 'x' shadows outer variable [shadowVariable]

FIX: Change name of the variable in or outside the scope

(style) Local variable 'y' shadows outer variable [shadowVariable]

FIX: Change name of the variable in or outside the scope

(style) Local variable 'z' shadows outer variable [shadowVariable]

FIX: Change name of the variable in or outside the scope