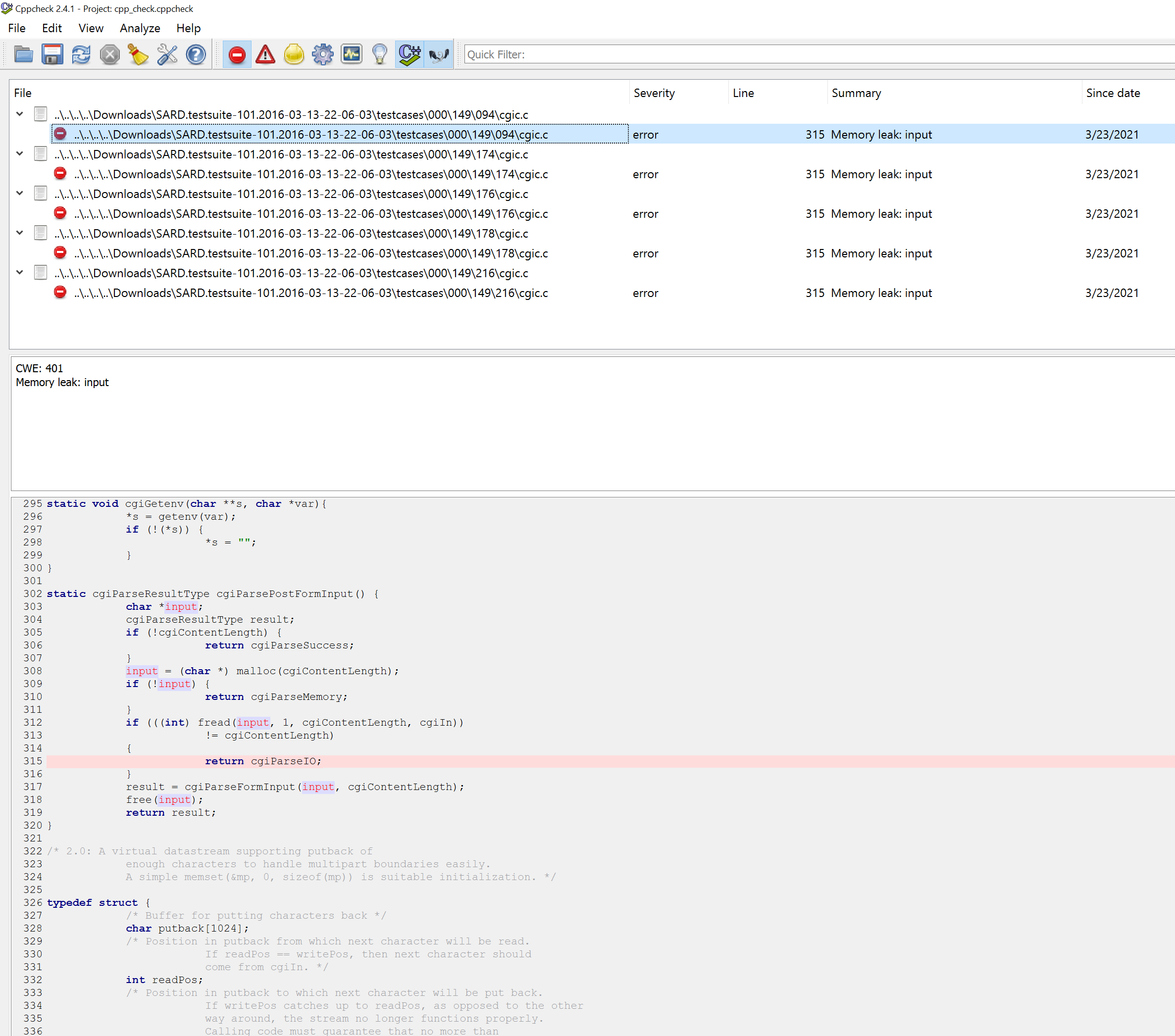
Michael Young (myoung10)

# Engagement 8-1

Here is the screenshot after following the instructions showing the memory leak. It seems that after calling malloc on line 308, they have a situation where they read into memory some content (cgiIn). After reading this, they return out of the function before calling the free function, which causes the memory leak. To fix this, I assume they’d have to place a free(input) command above line 315.



2nd issue.

The only other issue I could find using cppcheck was by toggling on ‘warnings’. The error filter showed the same example, and everything else blank with the exception of ‘style warnings’ which aren’t generally attackable. This issue is a possible null pointer dereference, which would crash the application.

In this situation, on line 908, if argValue is not set, then this will crash the assignment. However, after looking through the code, it seems that argValue is actually a char array, as I see APPEND statements prior to the assignment on 908. On line 878, they assign argValue to an integer value. My guess is that they should have set this initial value to NULL or ‘0’ enclosed in quotes, or (char)0. As well as possibly make this a pointer assignment. Also, having a null pointer check at line 907 as part of the conditional. If the caller were to make argValueSpace true, it would uncover this issue.

