

An advanced static analysis suite designed to uncover complex and elusive bugs that require extensive time and disk space to detect.

With a comprehensive collection of specialized checks, CodeCheck employs sophisticated algorithms and exhaustive analysis techniques to identify hard-to-find bugs within your codebase.

Bug Hunter's strength lies in its unwavering dedication to thoroughness, meticulously exploring every possible execution path and evaluating complex control and data flow scenarios. This comprehensive approach ensures the detection of bugs that often elude conventional analysis tools.

For extra accuracy, make sure to enable the AST Cache under Project - Configure Project - C++

Checks by Severity



Checks

| Check ID | Check Name | Supported | Severity |
|-----------------------------|--------------------------|-----------|----------|
| CPP_SA_DANGLING_POINTER_S | Dangling Pointer | Yes | High |
| CPP_SA_DIV_ZERO | Division by Zero | Yes | High |
| CPP_SA_LEAKS | Memory Leak | Yes | High |
| CPP_SA_NULL_PTR | Null Pointer Dereference | Yes | High |
| CPP_SA_STACK_ADDRESS_ESCAPE | Stack Address Escape | Yes | High |
| CPP_SA_UNDEFINED_CALL | Undefined Call | Yes | High |
| CPP_SA_UNINITIALIZED | Uninitialized Value | Yes | High |
| CPP_SA_VIRTUAL_CALLS | Virtual Call | Yes | High |