

Clang

The clang family of compilers is built on the LLVM compiler framework. It is an open source compiler supporting a BSD-like license. See <http://clang.llvm.org/docs/> for additional information. The following Clang compilers are supported:

- [Clang C/C++ Compiler v 3.9](#)
- [Clang C/C++ Compiler v 4.0](#)
- [Clang C/C++ Compiler v 5.0](#)
- [Clang C/C++ Compiler v 6.0](#)

Clang C/C++ Compiler v 3.9

- Compiler acronym: clang_3_9
- Host OS: Linux
- Supported languages: C89, C99, C++98, C++11, C++14
- Supported practices: Full support
- Support level: [Standard](#)

Additional Notes

- Only the x86_64 target has been tested. If the clang 3.9 compiler is built as a multi-target compiler, C++test code analysis may not function as expected on cross-compilation targets.
- The pre-built cpptest runtime is not supported for dynamic analysis (unit testing and application monitoring). The runtime must be built during the execution of the test configuration by setting `autoBuildDefault="true"` in the test execution flow.
- The xmm intrinsic headers are not supported for dynamic analysis.
- Clang is used to link inside of C++test by default. If errors occur during linking that relate to missing C++ libraries, then change the linker in the Build Settings from `clang` to `clang++`.
- The `clang_3_9` compiler configuration does not use the default GCC for selecting headers. As a result, problems can occur during compilation of instrumented source code during unit testing and application monitoring.

Clang C/C++ Compiler v 4.0

- Compiler acronym: clang_4_0
- Host OS: Linux
- Supported languages: C89, C99, C++, C++11, C++14, GNU++11, GNU++14
- Supported practices: Full support
- Support level: [Standard](#)

Additional Notes

- C++test does not support C++17 features accepted by clang
- Only the x86_64 target has been tested. If the clang 4.0 compiler is built as a multi-target compiler, C++test code analysis may not function as expected on cross-compilation targets.

Clang C/C++ Compiler v 5.0

- Compiler acronym: clang_5_0
- Host OS: Linux
- Supported languages: C89, C++11, C++14
- Supported practices: Full support
- Support level: [Standard](#)

Additional Notes

- By default, clang is used to link inside of C++test. If errors occur during linking that relate to missing c++ libraries, then change the linker in the build settings from `clang` to `clang++`.
- Only the x86_64 target has been tested. If the clang 5.0 compiler is built as a multi-target compiler, C++test code analysis may not function as expected on cross-compilation targets.
- C++17 is not currently supported.
- The `avx512` intrinsic headers are not supported
- This version has been tested with GCC 6 and newer. Pairing this compiler with older versions of GCC is unsupported.

Clang C/C++ Compiler v 6.0

- Compiler acronym: clang_6_0

- Host OS: Linux
- Supported languages: C89, C99, C++98, C++11, C++14, C++17
- Supported practices: Full support
- Support level: [Standard](#)

Additional Notes

- The `avx2` intrinsic headers are not supported.
 - The `avx512` intrinsic headers are not supported
 - This version has been tested with GCC 6 and newer. Pairing this compiler with older versions of GCC is unsupported.
-

About Support Levels

- **Extended:** Support has been validated with extended testing and is approved for use in safety-critical software development.
- **Standard:** Support has been validated with standard testing and is approved for use in non-safety critical software development.