Coverity on Polaris 2022.9.0 Supported Platforms, Languages, and Compilers

Introduction

Platform Support for Coverity on Polaris

Platform support info comes from the Coverity Installation and Deployment Guide, section 7.4 Coverity Analysis (platform support):

https://build-sig.internal.synopsys.com/job/DOC/job/doc-2022.9EN-packaging/lastSuccessfulBuild/artifact/build/latest/en/cov_deploy_install_guide.html#static_analysis_platform_support

Latest: https://build-sig.internal.synopsys.com/job/DOC/job/doc-mainEN-packaging/lastSuccessfulBuild/artifact/build/latest/en/cov_deploy_install_guide.html#static_analysis_platform_support(Notes in this box do not appear in the final documentation.)

Table 1. **Platforms Supported**

This support information applies to the installation of Polaris CLI Client and Coverity Analysis.

OS	Version	Notes
Linux	64-bit kernel, version 2.6.32+ with glibc 2.18~~-2.27~~ or later	**Linux Platform Support Notes** • Debian GNU/kFreeBSD is not supported.

macOS	OSX 10.15, 11, 12	• macOS 12 on Apple Silicon.
		  • Running Sigma on ARM-based Mac machines is not supported.
		  • Build capture is not currently supported.
		• Deprecation notice: Support for macOS 10.15 is deprecated as of 2022.6.0 and will be removed in a future release.

Windows	x86_64, Version 10 and 11 ~~later~~ and Windows Server 2019 and 2022	**Windows Platform Support Notes** • Coverity Analysis for C# and Visual Basic supports analysis of programs compiled by the Visual C# compiler (csc.exe) and Visual Basic compiler (vbc.exe) from .NET Framework versions 3.5 SP1 and 4.5.2–4.8. • Coverity Analysis is not supported on 'Server Core' installations at this time. • ~~Coverity tools run notably slower on Windows than on other operating systems, such as Linux.~~

- When performing Java code analysis, Coverity requires Oracle Java SE Runtime Environment 8 (JRE-8) as its executable platform. This requirement is unrelated to your choice of Java compiler.

Language Support for Coverity on Polaris

Unless otherwise noted, language support information (the following table) comes from the Coverity Analysis Guide, section 1.3 Language Support :

https://build-sig.internal.synopsys.com/job/DOC/job/doc-2022.9EN-packaging/lastSuccessfulBuild/artifact/build/latest/en/cov_analysis_administration_guide.html#language_support

And section 2.1 Simplified analysis support matrix (for Swift support on Sigma)

https://build-sig.internal.synopsys.com/job/DOC/job/doc-2022.9EN-packaging/lastSuccessfulBuild/artifact/build/latest/en/cov_analysis_administration_guide.html#supportMatrix.d12e4340

(Notes in this box do not appear in the final documentation.)

Table 2. **Language Support**This table describes languages supported for Coverity on Polaris and Coverity capture modes available for each language. Support might differ from standard Coverity.

Language	Capture mode	OS	Versions
Apex	• Buildless capture • Autocapture	Windows macOS Linux	
C/C++	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.)	Windows macOS Linux	• C++20 • C++98 • C++03 • C++11 • C++14 • C++17 • C89 • C99 • C11
C#	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.) • Buildless capture • Autocapture	Windows Linux	Up to C# 10
Objective-C	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.) • Autocapture	Windows macOS Linux	NA
Go	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.) • Autocapture	Windows macOS Linux	Go ~~1.16~~ 1.17
Java	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.) • Buildless capture • Filesystem capture • Autocapture	Windows macOS Linux	Up to Java 1~~7~~

JavaScript	• Buildless capture • Filesystem capture • Autocapture	Windows macOS Linux	ECMAScript 5
Kotlin	• Build capture (See compiler support table below. Build capture requires that your project use a compiler version which is supported by Polaris.) • Autocapture	Windows macOS Linux	Kotlin ~~1.5.20~~ –
PHP	• Buildless capture • Filesystem capture • Autocapture	Windows macOS Linux	PHP 7.0.0
Python	• Buildless capture • Filesystem capture • Autocapture	Windows macOS Linux	Python 3.x– 3.
Ruby	• Buildless capture • Filesystem capture • Autocapture	Windows macOS Linux	Matz's Refere equivalents
Swift	• Build capture (See compiler support table below) • Autocapture	macOS	Up to 5.7
Typescript	• Buildless capture • Filesystem capture • Autocapture		TypeScript 1.(
Visual Basic	• Build capture	Windows	Up to Visual B

IaC Platform and File Format Support

Infrastructure as Code (IaC) support uses autocapture in Polaris to detect security flaws in your infrastructure code in the following platforms and file formats.

Platforms

- AWS CloudFormation
- Kubernetes
- Terraform

Formats

- HCL (Terraform)
- JSON
- XML
- YAML

Compiler Support

Unless otherwise noted, compiler support information comes from the Coverity Installation and Deployment Guide, Chapter 8 Supported languages, compilers, and frameworks:

Latest: https://build-sig.internal.synopsys.com/job/DOC/job/doc-2022.9EN-packaging/lastSuccessfulBuild/artifact/build/latest/en/cov_deploy_install_guide.html#compilers

(Notes in this box do not appear in the final documentation.)

Compiler information is for Build Capture only

- Running Coverity with Build Capture requires that your project employ a version of the compiler which is supported by Coverity. In this section, each table lists the supported compilers and versions for the language in question. Note that compiler support is not identical between Coverity on Polaris and standard Coverity.
- If you don't plan to run Build Capture, compiler support is not relevant and the information in the Language Support table (just before this section) should be sufficient.

Table 3. **C/C++ Compiler Support**Build capture for C/C++ source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	FSF GCC 4.0~~–11~~. 12.1.0	**LLVM** • Clang compilers have various use limitations with Coverity products, which are listed in the Coverity Analysis User and Administrator Guide. • ~~Deprecation Notice: Support for LLVM Clang 5.0 is deprecated as of 2022.6 and will be removed in a future release.~~ **GCC** • ISO/IEC TR 18037 fixed point extensions are supported for C (not C++) code. • Versions of any of these compilers that are modified to accept non-standard syntax are not supported. • GNU GCC compilers distributed with Apple Xcode are not supported. **Android NDK Clang** • Clang 3.3 with Android NDK r9d is not supported, please use Clang 3.4 instead. **MSVC** • Managed C++ and Common Language Runtime (CLR) are not supported. Compilations with switches beginning with "/CLR" will be skipped. • Deprecation notice: Support for Visual Studio 2013 is deprecated as of 2022.6 and will be removed in a future release. • Deprecation notice: Support for Visual Studio 2015 is deprecated as of 2022.6 and will be removed in a future release.
macOS	LLVM Clang ~~5~~ 6.0–14.0 Android NDK Clang 3.1-3.4 (NDK revisions r8c-r9d) FSF GCC 4.0~~–11~~. 12.1.0	
Windows	FSF GCC 4.0~~–11~~. 12.1.0 Microsoft Visual C++ 2013-2022	

Table 4. **C# Compiler Support** Build capture for C# source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	.NET Core 3.1 .NET 6.0	• Linux 64-bit releases that support .NET 6. Coverity's C# compiler depends on an included .NET 6 runtime.
macOS	Not supported	
Windows	Visual Studio 2013, 2015, 2017, 2019 .NET Core 3.1 .NET 6.0	• Visual Studio Express editions are not supported. • Coverity supports analysis of Windows RT applications. • Deprecation notice: Support for Visual Studio 2013 is deprecated as of 2022.6 and will be removed in a future release. • Deprecation notice: Support for Visual Studio 2015 is deprecated as of 2022.6 and will be removed in a future release. • Deprecation notice: Support for .NET Core 3.1 is deprecated as of 2022.9.0 and will be removed in a future release.

Table 5. **Objective-C Compiler Support** Build capture for Objective-C source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	Not supported	
macOS	LLVM Clang 5.0–14.0	• Deprecation Notice: Support for LLVM Clang 5.0 is deprecated as of 2022.6 and will be removed in a future release.
Windows	Not supported	

Table 6. **Go Compiler Support** Build capture for Go source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	Go compiler ~~1.16~~ 1.17-1.18	• Coverity only supports projects that are built with the following commands: go build, go install, go run, and go test. Coverity does not support projects that are built by invoking either go tool compile or gccgo directly. • Coverity does not directly recognize custom flags and arguments of go run or go test. In order for Coverity to recognize these custom flags and arguments, you must modify config /templates/go/go_switches.dat. • The cov-emit-go command might have dependencies on external tools, depending on the Go code being compiled. Refer to the cov-emit-go command in the Coverity Command Reference for details. • Build capture is not currently supported on macOS on Apple Silicon. • Deprecation Notice: Support for Go 1.17 deprecated as of 2022.9 and will be removed in a future release. • ~~Deprecation Notice: Support for Go 1.16 deprecated as of 2022.6 and will be removed in a future release.~~
macOS	Go compiler ~~1.16~~ 1.17-1.18	
Windows	Go compiler ~~1.16~~ 1.17-1.18	

Table 7. **Java Compiler Support** Build capture for Java source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	~~Sun/~~Oracle JDK ~~1.7~~ 1.8, 11, 17, 18 OpenJDK 1.8, 11, 17, 18	• JDK 1.7 is not supported on Windows ~~10~~. • Coverity does not support Oracle JRockit JDK • macOS: To ensure a complete capture, version 11.x - 13.x of Xcode (with command line tools) must be installed. Please ensure `xcodebuild -version` runs without error; that command is used by Coverity to check the version. • macOS: Build capture is not currently supported on macOS on Apple Silicon. • Deprecation notice: Oracle JDK 18 is deprecated as of 2022.9.0 and support for it will be removed in a future release. • Deprecation notice: OpenJDK 18 is deprecated as of 2022.9.0 and support for it will be removed in a future release. • ~~Deprecation Notice: Support for Sun /Oracle JDK 1.7 is deprecated as of Coverity 2022.3.0 and will be removed in a future release.~~
macOS	~~Sun/~~Oracle JDK ~~1.7~~ 1.8, 11, 17, 18	
Windows	~~Sun/~~Oracle JDK ~~1.7~~ 1.8, 11, 17, 18 OpenJDK 1.8, 11, 17, 18	

Table 8. **Kotlin Compiler Support** Build capture for Kotlin source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	Kotlin ~~1.5.20 - 1.5.32~~, 1.6-1.6.~~10~~ 21, 1.7.0	• Coverity only supports Kotlin projects that are targeted to JVM or Android, not other platforms. For multiplatform projects, Coverity only captures Kotlin source files that are targeted to the supported platforms. • macOS: To ensure a complete capture, version 11.x - 13.x of Xcode (with command line tools) must be installed. Please ensure `xcodebuild -version` runs without error; that command is used by Coverity to check the version. • ~~Deprecation Notice: Support for Kotlin 1.5.x is deprecated as of Coverity 2022.3.0 and will be removed in a future release.~~ • Deprecation notice: Support for Kotlin 1.6.x is deprecated as of 2022.9.0 and will be removed in a future release.
macOS	Kotlin ~~1.5.20 - 1.5.32~~, 1.6-1.6.~~10~~ 21, 1.7.0	
Windows	Kotlin ~~1.5.20 - 1.5.32~~, 1.6-1.6.~~10~~ 21, 1.7.0	

Table 9. **Swift Compiler Support** Build capture for Swift source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	Not Supported	
macOS	IDE: Xcode 12.5.x	• Cross-compilation using Mac Catalyst is not supported. • Coverity Analysis supports Swift compiler invocations via xcodebuild. Swift Package Manager is not supported. • Build capture of Swift is supported internally with `cov-emit-text`. Autocapture must be disabled.
Windows	Not Supported	

Table 10. **Visual Basic Compiler Support** Build capture for Visual Basic source code is supported when the project uses any of the following compilers.

OS	Supported Compilers	Versions / Notes
Linux	Not Supported	
macOS	Not Supported	
Windows	Visual Studio 2013-2019 Compiler .NET Core 3.1	Compiler .NET Core 3.1 is for Visual Basic. NET. Visual Studio Express editions are not supported. Deprecation notice: Support for .NET Core 3.1 is deprecated as of 2022.9.0 and will be removed in a future release.

-