klocwork

PERFORCE

The Best Static Analysis and SAST Tool for Accelerating Time-to-Market and Delivering High Quality, Secure, and Compliant Code

Overview

Klocwork is a static analysis and SAST tool for C, C++, C#, Java, JavaScript, Python, and Kotlin. It identifies software security, quality, and reliability issues to help enforce standards compliance.

Built for enterprise DevOps and DevSecOps, Klocwork scales to projects of any size, integrates with large complex environments, a wide range of developer tools, and provides control, collaboration, and reporting.

Klocwork's Differential Analysis engine provides instant analysis results, while maintaining accuracy, and integrates seamlessly with CI/CD pipelines to automate Continuous Compliance — safeguarding your software from vulnerabilities with every commit.

Key Features

FIND SECURITY VULNERABILITIES WITH SAST

Our security-focused static analysis engine identifies security vulnerabilities as they are introduced – helping to find and fix vulnerabilities early, and provide compliance to internationally and industry recognized security standards, as well as your own organizational requirements.

PROJECT STREAMS

Project Streams provides easy management of shared code bases that have multiple variants or branches by simplifying project rule configuration, issue management, defect citing, reporting, and efficient data storage of analysis data.

Creating streams provides the following benefits:

- Assign a single project rule configuration to all variants.

- Issues common to multiple variants are automatically kept in sync and only require citing once.

- Easily identify identical issues across multiple streams and issues unique to a specific stream.

- Generate reports on individual streams for compliance, functional safety, or other evidential purposes.

- More convenient organization and efficient storage of analysis data.

DEVOPS READY

Klocwork tools are designed with Continuous Integration and Continuous Delivery foremost in our thinking, which makes it easy to include static code analysis as part of your CI/CD pipelines

Differential Analysis: Using system context data from the Klocwork Server, it is possible to analyze only the files that changed providing differential analysis results as if the entire system had been analyzed and the shortest possible analysis times.

Easy to Automate: Klocwork tools have common command line interfaces, and all defect data is accessible via a REST API using standard output formats, such as XML, JSON, and PDF.

Containerized Builds: Klocwork can be run within containerized and Cloud build systems and supports the provisioning of machine instances as required. Providing maximum flexibility and opportunity to use on-premise or external Cloud services for code analysis.

CONTROL, COLLABORATION, AND REPORTING

The Klocwork Validate platform is a centralized store of analysis data, trends, metrics, and configurations for codebases across the organization — accessed through a web browser.

The dashboard is highly customizable, enabling your developers, managers, and other stakeholders to:

- Define global or project-specific QA and security objectives and rule configurations.
- Control access permissions and approval workflows.
- View trending and metrics data for project quality and compliance.
- Produce compliance and security reports.
- Prioritize defects based on severity, location, and lifecycle.
- Use Smart Rank to assist developers in prioritizing fixes based on defect likelihood, which when combined with issue severity, provides an overall vulnerability risk score.
- Distinguish new issues from legacy code issues.
- Push backlog issues to Change Control systems.

DESIGNED FOR DEVELOPERS

By seamlessly integrating static code analysis with the rest of your development toolset, Klocwork will shift-left defect detection and improve developer adoption as a tool for developer training and increasing productivity.

No User Configuration: Klocwork provides out of the box support for hundreds of compilers and cross-compilers.

Easy to Use: Plugins for popular IDEs (including Microsoft Visual Studio, Eclipse, IntelliJ, and more).

Connected Desktop: Local code changes made using the Klocwork plugins provide immediate differential analysis results within IDEs.

Detailed Feedback and Help: Defects and coding violations are identified by severity, location and risk. Each defect report is further enhanced with detailed traceback information and rich, context-sensitive help and guidance on remediation. Facilitating understanding and learning.

In addition, Klocwork features a Secure Code Warrior integration, which provides you with software security lessons and training tools for many common development languages as you write code.

Custom Rules: A graphical custom checker creation tool makes the implementation of project- or organization-specific rules quick and easy — further enriching the learning opportunities.

Architectural Analysis: Klocwork integrates with architectural visualization and enforcement tools like Structure 101 to allow users to further improve the overall quality and maintainability of their codebase through clean and correct dependencies.

Technical Specifications

SUPPORTED LANGUAGES

- C
- C++
- C#

- Java
- JavaScript

- Python
- Kotlin

SUPPORTED FRAMEWORKS

C/C++

- AUTOSAR
- Boost
- Microsoft .Net
- POSIX
- QT
- STL
- WinAPI

C#

- .NET Framework
- .NET Core
- Mono
- Xamarin
- Unity
- Universal Windows Platform

Java

- Android
- Java SE/ EE
- Junit
- Hibernate ORM
- Apache Cocoon
- Apache Commons
- Apache ECS
- Apache Struts
- Apache Tomcat
- log4j
- Eclipse SWT
- JDOM
- Spring Framework
- GWT
- Java Persistence API

- JAX RS
- JAX WS
- ReactiveX
- Vert.x
- WS XML-RPC

JavaScript

- TypeScript
- JSX
- React
- Vue

Python

- Python 3

SUPPORTED CODING STANDARDS

Security:

- CERT (SEI)
- CWE (SANS)
- CWE (SANS) Top 25

- OWASP
- DISA STIG v5

- PCI DSS
- TS 17961 (ISO/IEC)

Safety:

- MISRA C 2004
- MISRA C 2012
- MISRA C 2023

- MISRA C++ 2023
- MISRA C++ 2008
- AUTOSAR C++ 14

- JSF AV C++
- TS 17961 (ISO/IEC)
- HKMC

Quality:

- NASA's 10 Rules

- Klocwork Quality

Custom:

- Create Your Own Standard

- Create Your Own Rules

SUPPORTED FUNCTIONAL SAFETY STANDARDS   *TÜV-SÜD certified for compliance.

- ISO 26262 (automotive) up to ASIL level D.*
- IEC 61508 (general industry) up to SIL 4.*
- EN 50128 (railways) up to SW-SIL 4.*

- IEC 62304 (medical devices) up to Software Safety Class C.*
- IEC 60880 (nuclear power).*

SUPPORTED PLATFORMS

- Windows
- Linux

SUPPORTED IDES

- CLion
- Eclipse
- Wind River Workbench
- Visual Studio Code
- Microsoft Visual Studio
- QNX Momentics
- Android Studio
- WebSphere
- IBM Rational Application Developer
- JetBrains Intelilj IDEA

SUPPORTED SOURCE CODE MANAGEMENT SYSTEMS

*Snapshot views are not supported for Base ClearCase
**Subversion 1.4.x is not supported by the Visual Studio plug-ins

- Base ClearCase 7.x*
- CVS 1.12.x
- Git 1.7.x
- TFS 2010
- Perforce server 2005.2 or higher
- Subversion 1.4.x**, 1.6.x, 1.7.x, 1.8.x

CRITICAL CHECKS

- API Usage Errors
- Dangerous Coding Practices
- Buffer Overflows
- Exposed Fields, Identifier Name Clashes
- Code Complexity
- Code Maintainability Issues
- Concurrent Data Access Violations
- Concurrency Issues
- Cross-Site Request Forgery (CSRF) Vulnerabilities
- Cross-Site Scripting (XSS) Vulnerabilities
- Dangerous Implicit Conversions
- Dead Code
- Error Handling Issues
- Hard-Coded Credentials
- Improper Certificate Validation
- Improper Encapsulation
- Incorrect Error Handling
- Indeterminate Value Warnings
- Information Leakage
- Invalid Arithmetic Operations
- Maintainability Issues

- Memory — Corruptions
- Memory — Illegal Accesses
- Missing Authentication for Critical Function
- Missing Authorization Checks
- No Configuration for a Critical Resource
- No Configuration for a Protect Resource
- Null Pointer Dereferences
- Object-oriented Programming Issues
- Path Manipulation
- Performance Issues
- Portability Issues
- Possible Runtime Failures
- Process and Path Injection
- Pseudorandom Number Generation Issues
- Redundant Code
- Resource Leaks
- Rule Violations
- Security Best Practices Violations
- Security Misconfigurations

- Stylistic Issues
- SQL Injections
- Suspicious Code Practices
- Suspicious Encapsulation
- Suspicious Scoping
- Uninitialized Members, Use of Uninitialized Fields and Variables
- Unnecessary Code
- Unreachable Code
- Unsafe Code Practices
- Unused Code
- Unused Local Variable
- Unvalidated User Input, Path/File/Process Injection, Tainted Data
- Use of Freed Resources
- Use of Hard-Coded Credentials
- Use of ldap Anonymous Bind
- Use of Weak Cryptographic Algorithm
- Vulnerable Coding Practices
- XML External Entity Attack
- XXE Vulnerabilities

SUPPORTED C/C++ COMPILERS

- Analog Devices Blackfin and TigerSHARC
- Archelon
- Archelon CSR Kalimba
- ARM CC
- ARM TI tms470
- CADUL C for Intel 80X86
- CEVA (NVIDIA)
- Clang
- Clang-cl
- CodeWarrior Freescale S12
- Compiler caching tools
- Cosmic
- Embarcadero
- Fujitsu FR
- GNU
- Green Hills
- GCC
- Hexagon Tools
- HI-CROSS+ Motorola HC16
- HI-TECH C
- Hitachi ch38
- HiveCC
- IAR 78K
- IAR 8051
- IAR ARM
- IAR Atmel AVR
- IAR AVR32
- IAR CR16C
- IAR Hitachi H8
- IAR M16C
- IAR M32C
- IAR MAXQ
- IAR MSP430
- IAR NEC V850
- IAR Renesas R32C

- IAR Renesas RX210
- IAR RH850
- IAR RL78
- IAR SH
- IAR STM8
- IBM XL
- ImageCraft AVR
- ImageCraft Intel
- ImageCraft M8C
- Intel iC-386
- Keil CA51, C166 and C251
- Marvell
- MetaWare
- Metrowerks CodeWarrior
- Microchip MPLAB C18
- Microchip MPLAB pic24
- Microchip MPLAB pic32
- Microchip MPLAB XC8 C
- Microchip MPLAB XC16
- Microsoft Visual Studio
- Microtec
- Microware Ultra C for OS-9
- Mono Headset SDK
- Motorola DSP563
- Nintendo Cafe Platform
- Nvidia CUDA
- NXP StarCore Freescale
- Panasonic
- Panasonic MN101E/ MN101L
- Paradigm
- Plan 9
- QNX qcc
- Renesas 78K0R
- Renesas CC-RL RL78
- Renesas CX
- Renesas M16C

- Renesas M32R
- Renesas R32C
- Renesas R8C
- Renesas RH850
- Renesas RX
- Renesas SuperH
- Renesas V850
- Rowley Crossworks MSP430
- Sony SN Systems PS2, PS3 and PSVita
- Sony Orbis Clang PS4
- Sun Studio
- Synopsys ARC MetaWare
- Target Chess
- Tasking 68K
- Tasking ARM
- Tasking Classic C166
- Tasking DSP56X
- Tasking IFX SLE88
- Tasking SLE88
- Tasking Tricore
- Tasking VX C166
- Tensilica Xtensa
- TI ARP32
- TI msp430
- TI tms320C55x
- TI tms320C3x
- TI tms320C4x
- TI tms320c28x
- TI tms320c6x
- TriMedia tmcc
- Watcom
- WinAVR
- Wind River Diab
- Wind River GCC
- ZiLOG eZ80

Try Klocwork Free

Get Started with your free trial of Klocwork today.

TRY FREE