CWE^{TM} COMPATIBILITY ENFORCEMENT

AUTOMATED SOURCE CODE ANALYSIS TO ENFORCE CWE COMPATIBILITY

STREAMLINE CWE COMPATIBILITY ENFORCEMENT

The Common Weakness Enumeration (CWE) compatibility enforcement module reports on dataflow problems, software defects, language implementation errors, inconsistencies and dangerous usage for C source code quickly and efficiently. The CWE C enforcement module is an optional add-on for the QA·C static analysis solution, providing a mapping of QA·C checks to CWE identifiers to ensure vulnerabilities including security related defects and violations are detected.

The CWE C enforcement module provides an extension to the analysis and reporting capabilities of QA·C to directly highlight known software vulnerabilities listed in the CW repository, and combines error detection and security best practice with full integration within the PRQA product suite.

The CWE C compatibility enforcement module provides an out-of-the-box configuration for QA·C, which eliminates the need to manually configure the tool to enforce CWE compatibility, and includes additional checks to supplement the already extensive suite of QA·C analysis checks. The existing QA·C report templates are also enhanced to allow generation of reports that specifically show the compatibility of a code base to the CWE database, to inform internal stakeholders or to use for audit purposes.

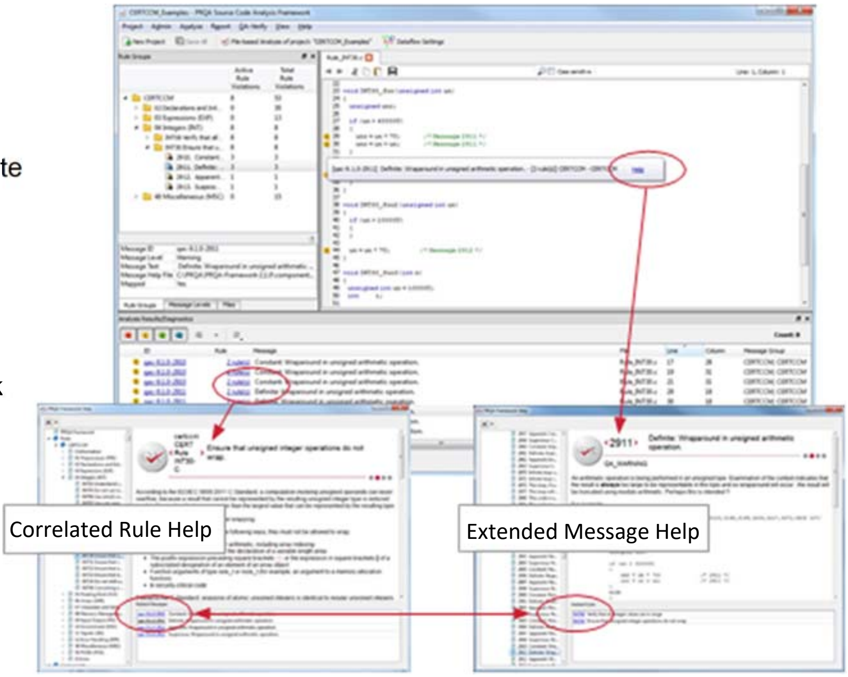
IDENTIFIES WHAT THE PROBLEM IS, EXPLAINS WHY IT'S A PROBLEM AND SHOWS HOW TO FIX IT

The QA·C static analyzer automatically performs in-depth analyses on your source code without executing programs. It checks your software for security vulnerabilities as described by CWE and can be configured to run locally on either desktop or server. QA·C identi- fies issues which compilers and most developers miss. These include lesser-known issues explicitly stated in the ISO standards and language constructs that, while not classified as incorrect, may result in unpredictable behavior.

Unlike bug catchers or less sophisticated static analyzers QA·C finds more issues while producing fewer false positives and negatives.

BENEFITS

- Automatically track, report and demonstrate CWE Compatibility.
- Continuously inspect source code for vulnerabilities in the CWE database
- Scale to millions of lines of code
- Increase code portability and re-usability
- Give your developers contextual feedback that helps them correct and learn from mistakes
- Reduce bottlenecks caused by manual code review and slow analysis tools and methods
- Analyze your source code without executing programs



DON’T JUST FIND BUGS - ENABLE BEST PRACTICE

CWE is a unified repository of known software weaknesses that have been shown to result in vulnerabilities that could be exploited. CWE, developed by the MITRE Corporation which is sponsored by US-CERT in the office of Cybersecurity and Communications at the U.S. Department of Homeland Security, provides a standard language for describing software security weaknesses. The standardization of terminology makes it easier for organizations to identify, understand and eliminate the countless security weaknesses that can occur in software. CWE is community-developed by a diverse, international set of experts from business, academic sources, software suppliers and government agencies, ensuring breadth and depth of content. The CWE enumerates design and architectural weaknesses, as well as low-level coding and design errors. It is not a coding standard but instead is a knowledge base of recognized software defects that are examples of insecure coding practices that should be avoided

For developers who lack security training, identifying security problems during a code review can be difficult, if not impossible. Security mistakes can be subtle and easy to overlook even for trained developers. The CWE C compatibility enforcement module plays a significant role in improving security and improving development practices. The use of this module can make the code review process faster and more effective by uncovering security related weaknesses and narrowing the set of potential problems for consideration during a code review.

KEY FEATURES

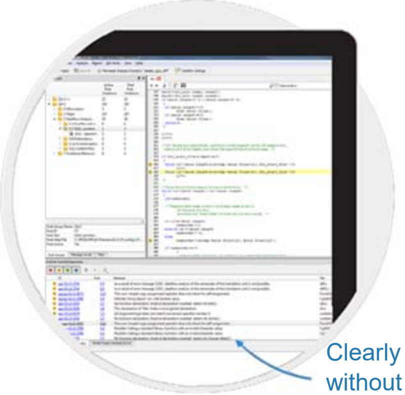
ADVANCED DEFECT PREVENTION

Using a proprietary, high-performance C language parser combined with a Deep Flow Dataflow analysis engine, QA·C is able to build an accurate model of the behavior of the software and track the value of variables in the code as they would be at run time. This sophisticated analysis approach maximizes code coverage while minimizing false positives and false negatives and allows QA·C to detect critical defects not reported by compilers or other tools and recognize issues caused by dangerous, overly complex and non-portable language usage.

Identify unpredictable behaviours others miss

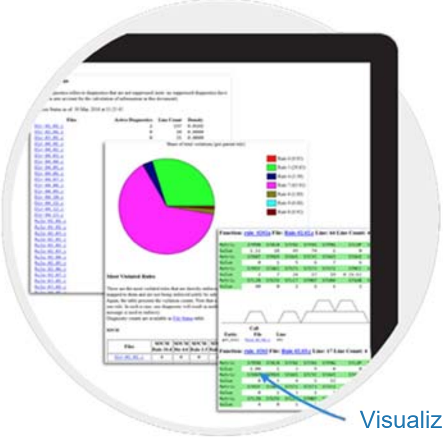
ACTIONABLE RESULTS TO COMPLY TO THE CWE STANDARD

The CWE C module clearly identifies must-fix defects and includes a comprehensive knowledge base help system that provides detailed guidance with examples to support developers in fixing the issues found in the source code. Because developers get immediate and contextual feedback within their development environment, they can make the required changes as they are creating new code or reviewing existing code. In this way, developers build aware- ness of best practice approaches and can quickly form coding habits that are aligned with your organization’s expectations.



Clearly identify errors without executing code

MONITOR AND CONTINUALLY IMPROVE YOUR CODEBASE WITH CONFIGURABLE REPORTS



Visualize what parts of the code need the most attention

The compatibility report helps you visualize which areas of your codebase require the most attention to reach a higher level compatibility.

The code review report refocuses peer review on discussing design, optimization, and meeting requirements rather than costly manual investigation of code conformance and correctness.

The suppression report provides information on message diagnostics that have been suppressed during analysis

ANALYSIS OF INDUSTRIAL-SCALE CODE

Automated static analysis using QA·C assists in identifying defects, vulnerabilities, and compatibility issues early in the development cycle where they can be fixed faster and at lower cost. QA·C is fast, non-disruptive, easy-to-use, and scales to any size of development environment. As a result, organizations whose products need to perform securely and reliably in mission critical and safety critical environments trust in QA·C to help lower the risk of software failures, improve quality and reduce time-to-market. The CWE C compliance module can ensure rapid and granular analysis of potential security issues both early in a development cycle and in established code streams by automatically detecting, reporting and providing context rich guidance on how to ensure security vulnerabilities are identified early makes it easier and more cost effective to fix.

EASY TO LEARN AND EASY TO USE

The CWE C module functions as a plug-in within QA·C's powerful GUI and delivers a contextual drill-down environment linked to a deep knowledge base. QA·C explains why problems it discovers need to be corrected and then provides guidance to help in fixing them.

ADAPTABLE TO FIT EXISTING DEVELOPMENT ENVIRONMENTS

The CWE C module plugs into QA·C and is easily integrated into existing build systems and continuous integration environments to provide a means to enhance “early and often” testing with automated code analysis that helps to avoid errors that are expensive to fix late in the development cycle. This allows existing code review processes to be accelerated and refocused, thereby helping to increase overall productivity while also improving quality and security of the software. Additionally, the CWE C module and QA·C can be configured for incremental analysis to ensure that only new changes are analyzed and feedback can be provided quickly.

ROBUST AND FLEXIBLE CODING STANDARD ENFORCEMENT

The CWE C module is based on the CWE online repository, to automate compatibility checks for the CWE weaknesses and the generation of the reports and audit documentation required to demonstrate compatibility. QA·C functionality also allows messages to be suppressed at targeted source code locations and these suppressions can be included in deviation reports when required for audit to a specific standard.

KEY CHECKS

The CWE C compatibility module helps to avoid constructs in the C language that can lead to product failures, functional safety issues and vulnerabilities that attackers can exploit and also reduce code reusability. The compatibility module applies the extensive QA·C message set supplemented by some additional CWE-specific checks to highlight weaknesses associated with the CWE identifiers. Documentation is provided describing rule enforcement and message interpretation, and an extensive set of example code is included to aid understanding.

The categories of vulnerabilities and weakness include:

- Boundary checks
- Resource leak checks
- Memory safety checks
- Dead code checks
- Uninitialized/unused variables checks
- Race conditions / synchronization checks
- Human coding errors

TECHNICAL SPECIFICATIONS

GENERAL FEATURES

- Command line interface (CLI)
- Interactive GUI with message browser
- Online help & knowledge base
    - Usage & implementation contextual message
    - C language
    - CWE compatibility
- Summary & detailed reports
- IDE integrations

CODE ANALYSIS FEATURES

- 1,700+ selectable messages
- C language-specific parsing engine
- Parses code of any size & complexity
- Handles common language extensions
- Cross module analysis (link time checking)
- Semantic error detection
- Dataflow error detection
- Close name analysis

MESSAGE OUTPUT CONTROL

- Comment based suppression
- Baselining

RESULTS OUTPUT

- Configurable HTML reports
- Standard report types
    - Compliance
    - Code review
    - Suppression
    - Metric data

CODING STANDARD ENFORCEMENT

- Identifies 120 CWE weaknesses, categories and compound elements
- CWE search - Users can search security elements using
- CWE identifiers
- CWE output - Security elements presented to users include, or enables users to obtain, associated CWE identifiers
- CWE documentation - Documentation describes CWE, CWE compatibility, and how CWE-related functionality is used
- Rule subsets for legacy code
- Best practice issues
- Naming convention checker
- Layout checker
- Defensive programming - defect avoidance
- Extensible rule base
- Customizable message text
- Deviation support

QA Systems and Programming Research Ltd

QA·C, QA·C++ and QA·Verify, offer the closest possible examination of C and C++ code. All contain powerful, proprietary parsing engines combined with deep accurate dataflow which deliver high fidelity language analysis and comprehension. They identify problems caused by language usage that is dangerous, overly complex, non-portable or difficult to maintain. Plus, they provide a mechanism for coding standard enforcement.

Contact Us

For further information regarding QA·C, QA·C++ and QA·Verify and compliance module add-ons, please contact QA Systems at **info@qa-systems.de.**