Introduction

- Scope
- Audience
- Usage
- How this Coding Standard Is Organized
- Relation to the CERT C Coding Standard
- Rules Versus Recommendations
- Tool Selection and Validation
- Conformance Testing
- Development Process
- System Qualities
- Automatically Generated Code
- Government Regulations
- Acknowledgments
- Automated Detection

This standard provides rules for secure coding in the C++ programming language. The goal of these rules is to develop safe, reliable, and secure systems, for example by eliminating undefined behaviors that can lead to undefined program behaviors and exploitable vulnerabilities. Conformance to the coding rules defined in this standard are necessary (but not sufficient) to ensure the safety, reliability, and security of software systems developed in the C++ programming language. It is also necessary, for example, to have a safe and secure design. Safety-critical systems typically have stricter requirements than are imposed by this coding standard, for example requiring that all memory be statically allocated. However, the application of this coding standard will result in high-quality systems that are reliable, robust, and resistant to attack.

Each rule consists of a title, a description, noncompliant code examples, and compliant solutions, as well as other information as described in How this Coding Standard Is Organized. The title is a concise, but sometimes imprecise, description of the rule. The description specifies the normative requirements of the rule. The noncompliant code examples are examples of code that would constitute a violation of the rule. The accompanying compliant solutions demonstrate equivalent code that does not violate the rule or any other rules in this coding standard.

A well-documented and enforceable coding standard is an essential element of coding in the C++ programming language. Coding standards encourage programmers to follow a uniform set of rules determined by the requirements of the project and organization rather than by the programmers’ individual preferences. Once established, these standards can be used as a metric to evaluate source code (using manual or automated processes).

This wiki contains ongoing updates of the standard between official published releases. If you are interested in contributing to these rules, create an account on the wiki and then request contributor privileges by sending a request to info@sei.cmu.edu.

The Secure Coding eNewsletter contains news from the CERT Secure Coding Initiative as well as summaries of recent updates to the standard rules. If you are interested in receiving updates directly, subscribe to the eNewsletter through our website or send a request to info@sei.cmu.edu.